

# 修 士 論 文 の 和 文 要 旨

研究科・専攻	大学院 情報理工学研究科 情報学専攻 博士前期課程		
氏 名	上野 正汰	学籍番号	1930015
論 文 題 目	ロボットによる調理作業間に必要な中間作業の研究		
<p>要 旨</p> <p>本研究の目的は、双腕ロボットシステムを用いて調理作業間に必要となる中間作業の実現によってサラダを作成する事である。これを達成するため、先行研究の調理作業での前提条件の考察、中間作業に必要な作業の検討、中間作業の実装、料理の作業記述の作成、ロボット動作を生成するシステムの構築という課題を抽出して、取り組んだ。まず、サラダのレシピから必要な調理作業を切断・計量・混合作業と抽出して、それぞれの先行研究における前提条件を考察した。中間作業に必要な作業を食材・容器の移動作業(bring)と調理済み食材の収集作業(collect)、環境の初期化作業(clean)と定めて考察し、実装方法を検討した。また、一連の調理作業を表現し中間作業を生成するための環境・知識情報を含んだ記述を定義して、多様な食材の特性と調理作業の組み合わせに対応可能な作業記述を提案した。中間作業の実装では、必要な動作とシステムの実装とともに、食材を扱うための特性の検討を行った。最後に提案した中間作業と作業記述でサラダを作成するための動作手順と環境情報、知識情報を記述・実行して、トマトのサラダを作成することに成功した。</p>			

令和2年度 修士論文

## ロボットによる調理作業間に必要な中間作業の研究

情報理工学研究科  
情報学専攻

学 籍 番 号 : 1930015

氏 名 : 上野 正汰

主任指導教員 : 工藤 俊亮 准教授

指導教員 : 末廣 尚士 教授

提出年月日 : 令和3年1月25日

# 目次

<b>第1章</b>	<b>緒言</b>	<b>1</b>
1.1	研究背景 . . . . .	1
1.2	関連研究 . . . . .	2
1.3	研究の目的 . . . . .	3
1.4	論文構成 . . . . .	4
<b>第2章</b>	<b>中間作業における課題の検討</b>	<b>5</b>
2.1	調理作業間に必要な作業に関する考察 . . . . .	5
2.2	作業の表現についての検討 . . . . .	7
2.3	食材の特性についての検討 . . . . .	8
<b>第3章</b>	<b>料理作業システム</b>	<b>10</b>
3.1	料理スクリプトの記述 . . . . .	10
3.1.1	食材・容器・調理器具の記述 . . . . .	11
3.1.2	作業の記述 . . . . .	15
3.2	中間作業の定義 . . . . .	16
3.3	中間作業の生成 . . . . .	18
<b>第4章</b>	<b>動作の実装</b>	<b>20</b>
4.1	システム概要 . . . . .	20
4.1.1	双碗ロボットシステム . . . . .	20
4.1.2	力覚センサ . . . . .	21
4.1.3	平行二指ハンド . . . . .	22
4.1.4	エンドエフェクタ . . . . .	22

4.2	食材の配置 . . . . .	25
4.2.1	食材・容器の移動作業 . . . . .	25
4.2.2	食材集め・移し替え作業 . . . . .	26
4.3	環境の初期化 . . . . .	30
4.4	調理作業の実装 . . . . .	31
<b>第5章</b>	<b>実験</b>	<b>32</b>
5.1	切断済み食材の移動実験 . . . . .	32
5.1.1	実験方法 . . . . .	32
5.1.2	結果・考察 . . . . .	34
5.2	サラダ作成実験 . . . . .	38
5.2.1	実験方法 . . . . .	38
5.2.2	結果・考察 . . . . .	42
<b>第6章</b>	<b>結言</b>	<b>51</b>
6.1	まとめ . . . . .	51
6.2	今後の展望 . . . . .	52
<b>参考文献</b>		<b>54</b>

# 目次

1.1	世界のロボット家電・コンシューマ向けロボット市場規模の推移及び予測	1
3.1	座標系の設定例	14
4.1	双碗ロボットシステムの概観	21
4.2	ロボットのベース座標系	21
4.3	力覚センサ	22
4.4	包丁	23
4.5	スプーン	23
4.6	泡だて器	24
4.7	物体把持エフェクタ	24
4.8	食材集め動作の座標系とパラメータの定義	28
4.9	移し替え動作の座標系とパラメータの定義	29
4.10	ふき取り動作	30
5.1	実験環境	34
5.2	使用器具	34
5.3	きゅうりの小口切りの結果	35
5.4	切断済み食材の移動実験の動作過程	37
5.5	実験環境	39
5.6	トマトの事前処理	39
5.7	サラダボウル(左)とドレッシング用のボウル(右)	40
5.8	醤油と酢の容器	40
5.9	清掃作業後のキッチンペーパー	43

5.10	ドレッシングの作成結果 . . . . .	43
5.11	サラダの作成結果 . . . . .	43
5.12	サラダ作成実験(トマトの切断作業) . . . . .	45
5.13	サラダ作成実験(酢と醤油の計量作業) . . . . .	47
5.14	サラダ作成実験(ドレッシングの混合作業) . . . . .	48
5.15	サラダ作成実験(ドレッシングの計量作業) . . . . .	50

# 表目次

5.1	実験に用いた食材特性 . . . . .	33
5.2	実験結果 . . . . .	35
5.3	レシピに関する食材情報 . . . . .	41
5.4	レシピに関する容器情報 . . . . .	41
5.5	レシピに関する調理器具情報 . . . . .	41
5.6	入力した料理スクリプト . . . . .	42

# 第1章

## 緒言

### 1.1 研究背景

総務省令和2年版情報通信白書によると、家庭における家事負担の軽減を目的としたロボット家電やコンシューマ向けロボットの世界市場(図 1.1)は年々拡大を続けている。2022年には2019年の2倍以上の市場規模となることが予想されている [1]。

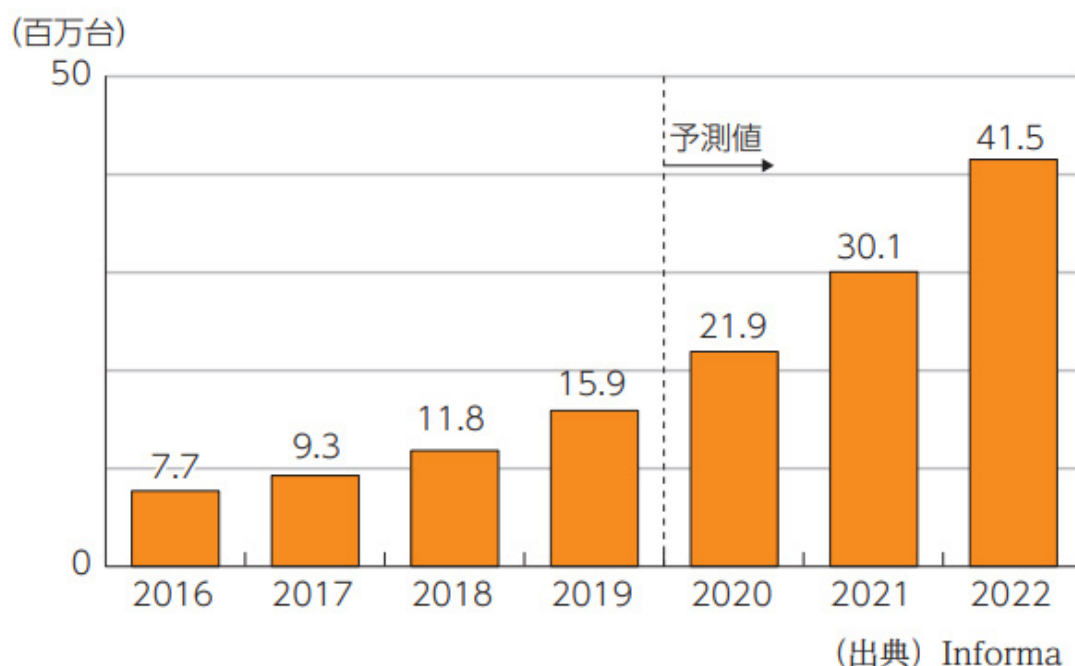


図 1.1: 世界のロボット家電・コンシューマ向けロボット市場規模の推移及び予測

市場規模の拡大とともに、近年家庭用ロボットは盛んに研究・開発が行われている。代表的な家庭用ロボットとしてiRobot社製のロボット掃除機「Roomba」が挙げられる。こ



れは自動で床を清掃するロボットである．このように，床掃除や窓ふきなど専門性が高く，単一の作業を行う家庭用ロボットは商用化されている．しかし，より複雑な作業や家庭における家事を汎用的に行うロボットは未だ商用化に至っていない．

人間は，抽象的な作業指示が与えられると，取得した知識や技術と環境情報から具体的な作業手順，動作を決定している．ロボットにおいても，事前にライブラリとして登録された操作スキルを入力された知識情報，環境情報を基に対象物に対して適応させて実行させることで，抽象的な作業記述を基に作業を実行できると考えられる．ロボットに適した抽象的な作業指示書を生成することができれば，人間の動作をロボットに置き換える効率が向上する．

抽象的な記述を基に作業を行うものとして，本研究室では料理に着目した．料理はレシピとして一般に普及している作業記述が存在している．また，料理で実行される作業は多岐に渡り，作業で扱う必要のある食材や道具類も膨大な種類が存在する．これらの点から，ロボットが抽象的な作業記述から実際に作業を実現するための研究課題として適していると判断した．本研究では，ロボットによる複数の調理作業を必要とする料理の実現を目標とする．

## 1.2 関連研究

料理作業を主題とした先行研究では，渡辺ら [2]は，ビジョンと力覚センサに基づく認識を利用して操作過程の状況を把握し，包丁操作を実現している．包丁を扱う動作は事前の作り込みであり，包丁の切断操作はセンサを用いて制御しているものではなかった．他にも，Schenckら [3]によるRGBD画像と熱画像を用いたお湯の注ぎ作業や，吉田ら [4]による生卵のかき混ぜ動作が挙げられる．

本研究室では，料理作業で実行される多様な調理作業をそれぞれ単一の調理作業として実現し，それらを連続で実行して一連の料理作業を実現することを目標として研究を行っている．各調理作業はそれぞれの機能をモジュール化し，完成度を高めていくことによって各調理作業をより一般的に実現している．今井ら [5]は，再利用性の高い一般的な切断作業の実現を目標として，力覚センサを用いて動的に制御を行いながら食材を切断するロボットシステムを実現している．董ら [6]は，RGBD画像によるフィードバックを用いたピーラーによる皮むき作業を実現している．

しかし、これらの単一の調理作業は開始状態や終了状態について個別の条件を設定している。今井らによる切断作業は包丁をアームに装着し食材をまな板の上に乘せた状態から、切断の動的制御を行って食材の切断を実行し、食材の切断が終了した時点で切断作業を終了している。ここで、料理作業を実現することを目的として考えると、切断作業を終了した状態から次の調理作業を開始するためには、次の調理作業で前提とされる状態に初期化する作業が必要となる。

本研究では、この調理作業間に必要となる作業を中間作業と定義し、料理作業を調理作業と中間作業の連続としてとらえる。そして、この中間作業を実現することで、先行研究の単一の調理作業の連続による一連の料理作業の実現を目指す。

また、料理のレシピを作業記述として扱う先行研究としてGravotらの取り組み [7]がある。この研究では、レシピを最上位の抽象的なタスクとして設定し、これをより具体的なサブタスクに分解していくタスクプランニングと、分解されたサブタスクをヒューマノイドロボットで実行するための動作計画を行っている。しかしこの取り組みはシミュレータ上で完結している。これに対して、福島ら [8]は人間用のレシピを解釈した動作手順から、事前知識や環境情報を用いて動作記述を生成し、液体や粉体を対象食材として計量・混合作業を実行することでドレッシングを作成している。福島らの動作生成システムでは、食材や容器、調理器具の表現が簡易であり、食材の様々な性質とその変化や容器に入っていない食材への作業や、細切りと輪切りなど単一の調理作業から実現できる作業の表現ができない。また、調理作業間で必要とされる動作は動作生成エンジンの一部として実装されるため、多様な調理作業に対して必要な動作を追加するたびに、動作生成アルゴリズムを変更する必要がある。

本研究では、食材・容器・調理器具の記述を拡張することで、様々な状態の食材を扱う多様な調理作業からなる料理作業を表現する。また、中間作業をモジュール化することで、多様な調理作業に対する対応を目指す。

### 1.3 研究の目的

調理作業間で前提とされている中間作業を抽出し、一連の料理作業を実行するシステムを構築することを本研究の目的とする。この目的を実現するにあたって次のような課題を解決する必要がある。

- 調理作業間に必要となる中間作業の検討
- 料理作業を具体的な動作に落とし込む記述の検討
- 必要な動作と作業の記述を実行するシステムの実装

まず、先行研究で実現されている調理作業における作業の開始時点と終了時点の前提条件から、必要となる中間作業を抽出する。中間作業の課題抽出は再利用性の高い作業とするために必要となる調理作業に限らず、本質的な課題を設定する必要がある。複数の調理作業が必要となる料理のひとつとしてサラダを作成することを具体的な課題として設定することで、中間作業の課題を抽出することとする。次に料理作業を具体的な動作に落とし込むための記述の検討を行う必要がある。作業の記述によって調理作業間の前提条件を明確にすることで、中間作業に必要な動作を絞ることができる。最後に検討した中間作業に必要とされる動作と作業の記述を実行するためのシステムを双碗ロボットシステム上で実装する必要がある。実験では、システムをサラダの作成に適用させ、提案した手法で実際にサラダが作成できるか確認する。

## 1.4 論文構成

本稿の構成を以下に示す。第2章ではサラダ作成に必要な調理作業から必要な中間作業を検討する。第3章では、システムの具体的な実装方法を検討する。第4章は提案した動作の実装と先行研究の調理作業の再実装を行う。第5章では提案した手法を検証するための実験を行う。第6章はまとめ、および今後の展望である。

## 第2章

# 中間作業における課題の検討

先行研究によって単一の調理作業と食材の移動を伴わない単純な料理は実現されているが、複数の調理作業を含んだ料理作業では、調理作業で前提とされる中間作業が必要である。本研究ではロボットによるサラダの作成を通じて、先行研究における各調理作業の課題を解決する。

本章ではサラダ作成に必要な複数の調理作業から中間作業を考察し、課題と実装方法を検討する。

## 2.1 調理作業間に必要な作業に関する考察

サラダは野菜などの具材に塩、酢、油などの調味料を組み合わせたドレッシングをかけて盛り付けた料理の総称である。使用される具材は多岐に渡るが、一般的には主な具材は食べやすい大きさに切断した野菜を用いる。また、ドレッシングは一般に複数の調味料を適量ずつ組み合わせることで作成される。

そのため、サラダの作成には次の3つの調理作業が必要となる。

- 切断
- 計量
- 混合

これらの調理作業の先行研究では共通した前提条件として、ロボットの関節可動域の問題から作業領域への食材配置と、調理に用いる調理器具を把持が必要となる。また、調理作業を連続的に実行するためには、次の調理作業の準備のために食材を作業領域から片づ

ける必要がある。さらに次の調理作業に影響を与えないために、調理作業によって生じた作業環境の変化を初期化する必要がある。

よって中間作業に求められるのは以下の作業となる。

- 調理作業の前後での食材の配置(対象食材と生成物)
- 環境の初期化
- 調理器具の把持

中間作業で実行すべき作業は直前の調理作業の終了時の状態と次の調理作業の前提条件から決定される。対象食材の配置を例に挙げると、直前の調理作業の対象食材と次の調理作業の対象食材が同じであれば、すでに対象食材の配置は完了しているため、動作を実行する必要がない。つまり、中間作業は前後の調理作業の情報によって生成される。

ここで、調理器具の把持に関しては、調理作業中には器具に様々なベクトルで力が加わることが考えられ、安定して器具を把持することは難しい課題となる。しかし、器具を安定して把持するためにはエンドエフェクタや治具の改良が主題となるため、本研究の目的とは異なる。そのため、本研究では調理器具の把持動作は人の手によって実現し、調理作業の前後での食材の配置と環境の初期化に着目して中間作業を実現する。

環境の初期化は、直前の調理作業によって作業環境が変化し、次の調理作業の実行に影響を及ぼす場合にそれを初期化することが求められる。そこで、サラダの作成に必要な切断、計量、混合作業の先行研究における調理作業で生じた作業環境の変化を調査した。すると、それぞれの研究では調理作業で扱った食材によって、作業領域の汚れが生じていた。作業領域が汚れた状態で次の調理作業を実行すると、意図せず容器に汚れが付着することや調理作業そのものに影響を及ぼすことが考えられる。そこで、本研究では環境の変化のひとつとして、作業領域の汚れに着目し、これを初期化するための作業を提案する。以降、この初期化作業をcleanと呼称する。

食材の配置では、直前の調理作業の終了時点での各食材の位置から次の調理作業の前提条件とされる食材の位置に配置することが求められる。これを実現するためには、まず食材や容器を作業環境上で移動するための作業が必要である。また、調理済み食材が作業領域に散らばっている状態の場合には、直前の調理作業の内容によって食材が様々な特性を持つため、平易な課題ではない。しかし、それぞれの中間生成物に対して容器を設定して一時的に収容することで、ハンドで直接扱うことが難しい調理過程の食材も扱うことがで

きる．そこで，食材の配置では，対象食材の移動に加えて生成物を収容するための容器の移動を行うこととする．

つまり，調理済み食材を収集して容器に収容する作業と対象食材と生成物を入れる容器を作業領域と非作業領域間で移動するための作業が必要となる．以降，容器に収容する作業をcollect，食材・容器を移動する作業をbringと呼称することとする．ここで，食材はその性質によって適した保存方法があり，液体や粒体，ペーストなどの状態の食材は保存容器に入れられており，野菜類やいも類などの大きな固体の食材の場合には，作業環境にそのまま置くことも考えられる．そのため，本研究では，各食材はそのまま置いてある状態と容器に入った状態が存在することとする．

bringの機能は食材の移動であるため，移動対象の食材の情報と目標の座標を入力して食材移動を行う必要がある．ここで，bringは食材の保存方法に依存せずに実行されるべきである．そのため，食材と容器の記述にはそれぞれの内包関係が記述されるべきである．また，次の調理作業で対象食材に対して特殊な配置方法や事前処理が必要な場合には，中間作業の機能としてではなく，調理作業側のモジュールに実現されるべきである．中間作業における食材の配置作業の目的は，次の作業を問題なく実行できるように食材を配置することである．

## 2.2 作業の表現についての検討

人間用のレシピには曖昧な表現が多く，記述されている内容だけでロボットの動作に落とし込むことは難しい．人間用のレシピは人が読んだときに補完可能な冗長な動作の説明を省くことで，手順を一般化して普遍性を高めている．これにより，人間はより少ない文字情報から料理作業を理解することができている．それに対して，高度に一般化された人間用のレシピをロボットが直接動作に落とし込むことは難しい．そのため，ロボットで複数の調理作業を含む料理作業を実現するためには，一連の作業をロボットが解釈できるように表現する必要がある．

調理作業を，単一の対象食材に対して操作を行って生成物を得るものとする．一連の料理作業を調理作業の連続として考えると，各調理作業で実行する作業と対象，生成物，使用する調理器具を記述することで，調理作業の手順を具体化することができる．しかし，調理作業の手順を具体化しても，調理作業で生成される中間生成物を調理過程での処理方

法や、環境の初期化作業などの情報はレシピには記述されていない。

連続した調理作業から作業間で必要となる中間作業を決定して、ロボットの動作に落とし込むシステムが必要となる。

## 2.3 食材の特性についての検討

料理に使用される食材は様々な特性を持ち、各食材をロボットで扱うことは平易な課題ではない。料理に使用される食材を考察するために、厚生労働省が定める国民健康・栄養調査食品群の大分類 [9]を1-10まで抜粋し、大分類に含まれる小分類の先頭に記載された食材を一部抜粋して以下に示す。各食材を物体としての特性でとらえると、穀類、砂糖・甘味料類には小さな粒体が含まれ、豆類や種実類はそれよりも大きな粒体が主である。対して野菜類や果実類、魚介類は様々な形状をした特性の異なる食材が存在する。加えてニンジンなど、ひとつの食材に絞って考えても、個体差や調理による形状や性質の変化などで様々な特性を持つことが考えられる。

1. 穀類（玄米，薄力粉，そば粉，とうもろこしなど）
2. いも類（さつまいも，じゃがいも，きくいもなど）
3. 砂糖・甘味料類（黒砂糖，和三盆糖，上白糖など）
4. 豆類（乾燥国産大豆，木綿豆腐，糸引き納豆，乾燥あずきなど）
5. 種実類（アーモンド，麻の実，えごま，カシューナッツなど）
6. 野菜類（トマト，葉にんじん，ほうれんそう，キャベツ，きゅうりなど）
7. 果実類（いちご，いよかん，バナナ，りんご（皮むき）など）
8. きのこと類（えのきたけ，乾燥きくらげ，干しいたけ，ぶなしめじなど）
9. 藻類（干しあおさ，青のり，干しのり，焼きのりなど）
10. 魚介類（まあじ，からふとます，あかがい，あかいか，あまえびなど）

これらの食材すべてに対して網羅的に実行できる一般的な動作を生成することは難しい。一部はリアルタイムなセンシングによって調整できるパラメータもあると考えられる。しかし、非破壊的に調べられない剛性などの特性やモデル化することが難しい課題も多い。

食材特性によってはロボットで実行する必要のある作業が変化することが考えられる。

水気の多い食材などを扱った際には作業する領域が汚れて次の調理作業に影響を及ぼすことが考えられる。また、ニンジンが一本ある状態と乱切りにされた状態とみじん切りにされた状態とでは、物体としての性質が大きく異なるため、制御する方法を変化させる必要がある。

そこで、食材の特性を大まかに分類しそれらを参照することで、多様な食材特性に対応できるロボットシステムを構築する。



## 第3章

# 料理作業システム

本章では，調理作業の連続からなる料理作業の記述と中間作業の生成アルゴリズムについて述べる．

### 3.1 料理スクリプトの記述

人間用のレシピをロボットが実行可能な動作手順として解釈するためには，連続する作業を表現するための記述が必要となる．2.2節で述べたように，人間用のレシピは冗長な説明省いて一般化するために，記述されていない要素が存在する．その一つとして，レシピの手順の記述の中には調理作業間に生成される中間生成物をどのように処理するかは記述されていない．人間が実際にレシピを見て料理をするときは，動作手順や環境に応じた生成物を容器に入れるなどの動作を人間が適切な解釈を行うことで実行する．

ロボットで各調理作業を実行するためには人間用のレシピよりも具体性の高いスクリプトが必要となる．そこで，調理作業間の影響無く作業を実行可能にするために，レシピを具体化した料理スクリプトを定義する．料理スクリプトは単一の調理作業の記述の連続とすることで，多様な調理作業に対してより一般的に表現することができる．各調理作業は実行する調理作業の情報に加えて，対象となる食材や生成物を入れる容器，作業に使用される調理器具の情報によって表現する．

本節では単一の調理作業を記述するための食材，容器，調理器具の記述を示す．

### 3.1.1 食材・容器・調理器具の記述

本項では、調理中に使用される食材・容器・調理器具を表現するための記述を提案する。各要素をロボットで制御するためには物体の位置姿勢や性質などの環境情報とそれぞれの制御に用いる知識情報が必要となる。そして、それらの情報は作業によって変化するため、作業間で統一して管理するモデルが必要となる。以下に食材・容器・調理器具を表現するための記述を示す。

- 食材(Ingredient)
  - name 食材の名称
  - size 食材の一般的な寸法
  - type 物体の状態(固体, 液体, 粒体, ペースト)
  - grasp\_force 食材に加えられる一般的な把持力
  - grasping\_coordinate 食材を把持するための座標系
  - character 食材特性や調理作業の詳細に対応する情報
  - pose 現在の食材座標系
  - initial\_pose 食材座標系の初期値
  - container 食材を入れるための容器
- 容器(Container)
  - name 容器の名称
  - size 容器の寸法
  - grasping\_coordinate 容器を把持するための座標系
  - handling\_coordinate 容器を制御するための座標系
  - pose 現在の容器座標系
  - initial\_pose 容器座標系の初期値
  - ingredient 現在入っている食材

- 調理器具(Tool)

- name 器具の名称
- size 器具の寸法
- handling\_coordinate 器具を制御するためのツール座標系の変換行列
- pose 現在の器具座標系
- initial\_pose 器具座標系の初期値

以上の要素を各調理作業間で共有して実行する。上記以外の情報が必要になった場合には、新しいメンバ属性を追加することで対応する。

これらの情報は、寸法や座標系の定義などの作業前に設定可能な知識情報と物体の位置などの作業環境に依存する環境情報に分けられる。知識情報はデータベースとして事前に用意して、料理スクリプトの生成時にnameによって参照する。環境情報は作業環境によって変化するため、料理スクリプトが生成される度に入力する必要がある。

poseは各物体の現在の位置・姿勢を逐次入力する。これにより、各物体の参照を調理作業間で共有することによって、各物体の座標系を統一して管理することができる。initial\_poseは物体の移動を実行した後で、初期位置に戻すために使用する。

grasping\_coordinate, handling\_coordinateには対象の物体を制御するのに有効な座標系を設定する。図 3.1にそれぞれに設定する座標系の例を示す。図のように、食材と容器はそれぞれの基準となる食材座標系、容器座標系上で把持点の座標系を設定する。これは位置姿勢情報から把持点の座標系を用いて手先座標系を合わせるために定義する。また、容器と調理器具は把持点の座標系を基準としたときの、制御に有効な座標系への変換行列を設定する。これは物体を把持した状態で制御するときには有効な座標系を定義する。

調理の過程で食材を入れるための容器をingredientのメンバ属性であるcontainerに設定する。料理スクリプトを実行するときには中間生成物を入れるための容器を参照することができる。これに対して、containerのメンバ属性であるingredientには現在収容されている食材を逐次設定する。これにより、スクリプトを実行するときには食材がすでに容器に入っているかを、作業間で統一して管理することができる。

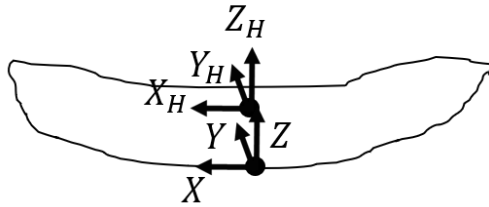
食材の物理的特性や調理作業の詳細な情報を格納するメンバ属性としてcharacterを設定する。2.3節で述べたように、料理では様々な特性を持つ食材を扱う必要がある。それぞれの特性に名前をつけてcharacterの中に紐づけて登録することで、食材特性の表現を

実現する。本研究において、きゅうりに登録したcharacterを以下に示す。characterに登録されている食材特性に関しては4において詳細を後述する。以下のように、食材の特性に名前を紐づけて属性値として参照することで、特性の記述を行う。また、調理済み食材の場合には調理作業の名前を紐づけることで、階層的に食材特性を表現することができる。

ソースコード 3.1: きゅうりのcharacter

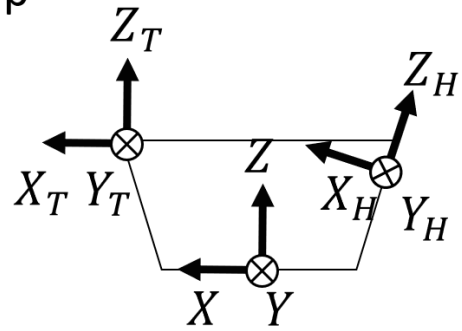
```
"cucumber": {  
  "character": {  
    "height": 0.038,  
    "wetness": 0,  
    "flexibility": 0,  
    "brittleness": 0,  
    "小口切り": {  
      "height": 0.001,  
      "wetness": 1,  
      "flexibility": 1,  
      "brittleness": 0,  
      "cut_length": 0.001  
    },  
    "乱切り": {  
      "height": 0.021,  
      "wetness": 1,  
      "flexibility": 0,  
      "brittleness": 0,  
      "cut_length": 0.02  
    }  
  }  
}
```

$H$ :Handle



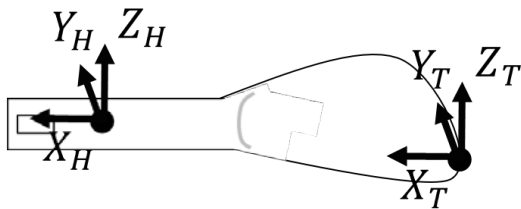
(a)食材の座標系の例

$H$ :Handle  
 $T$ :Tip

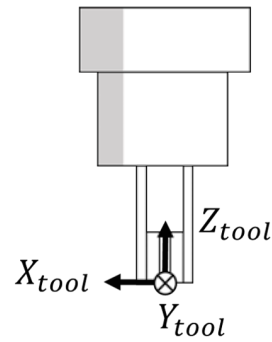


(b)容器の座標系の例

$H$ :Handle  
 $T$ :Tip



(c)調理器具の座標系の例



(d)ツール座標系の例

図 3.1: 座標系の設定例

### 3.1.2 作業の記述

本項では、単一の調理作業を表現するための記述を提案する。調理作業は対象の食材(target), 生成される食材(product), 使用する調理器具(tool)と各調理作業のシステム(cook\_direction)によって構成される。食材0をtool0で輪切りにして食材1を生成するときの作業記述の例を以下に示す。

ソースコード 3.2: 調理作業の記述例

```
{
  "target": ing[0],
  "product": ing[1],
  "tool": tool[0],
  "cook_direction": {
    "name": "輪切り",
    "func": cut
  }
}
```

以上のように単一の調理作業の記述には、作業間の独立性を保つために各調理作業を実行するのに必要な情報のみを記述する。これを連続的に実行していくことで料理スクリプトを実現する。

詳細は4.4節で後述するが、調理作業は以上のtarget, product, toolの情報を用いて実装する。多様な特性を持つ食材に対して様々な調理作業を実現するために必要となる詳細なパラメータは、対象食材のcharacterからcook\_directionのnameで参照することで設定する。

## 3.2 中間作業の定義

前節で提案した料理スクリプトを実行するためには調理作業間に必要な中間作業が必要となる。中間作業は前後の調理作業によって生じた生成物や対象食材によって必要な動作が変化する。動作の詳細な実装については4章にて後述するが、2.1節で述べた中間作業に必要な作業を以下に示す。

- *bring(ingredient, destination)*  
食材を目的の位置に移動する
- *collect(target, product, board, tool)*  
調理済み食材を容器に集める
- *clean(workspace\_pos, tool)*  
作業領域を清掃する

*bring*は入力した*ingredient*を*destination*に移動するための作業である。引数として入力する*ingredient*には食材の情報と食材を入れる容器の情報が格納される。それらを用いることで、移動作業を抽象化したまま記述する。食材が容器に入っていない状態である場合には、食材把持の座標系と把持力を用いて制御を行う必要がある。対して、容器に入っている食材の場合には、容器の縁を把持して持ち上げると内容物が零れることや、把持力が不足容器を落とすことが考えられる。そのため、容器に入った食材の場合には容器の縁を持ちながら引きずって移動する制御が必要となる。食材が中間生成物で容器に食材が入っていない場合には容器を持ち上げて移動させる。

*collect*は調理した*target*を*product*に紐づけられた*container*に入れるための収集運搬作業である。調理作業は容器内で実行されることもあるが、切断作業や塗布作業など調理作業によっては食材が作業領域に置かれた状態で終了することがある。*bring*で食材を移動させるためには、それらで生じた中間生成物を容器に収容するための作業が必要となる。

*collect*では調理済み食材に対して変形などの影響を与えずに、収集運搬することが求められる。しかし、調理では剛体に近い物体、やわらかい物体、容器に入った粒体や液体など様々な食材を使用することが考えられる。特に調理済みの食材は様々な特性をもつことが考えられる。切断作業の場合は対象食材が切断されて複数の小片となり、食材の種類や

切り方の組み合わせによっては、それらをハンドで把持することは難しい。そのため、多様な特性を持つ食材に対して収集作業を実行可能である必要がある。

そこで本研究では、ハンドで移動用の器具を把持して器具を食材の下に滑り込ませて把持することで、多様な特性を持つ食材に対してロバストな収集運搬作業を実現する。collectは移動用の器具に調理済み食材を乗せる食材集め動作と移動用器具から移動先の容器へ食材を移す移し替え動作で構成する。また、collectでは食材移動によって容器の中に食材が入るため、移動先のcontainerのingredientを更新する必要がある。

ここで、食材集め動作では食材の下に器具を滑り込ませ、移し替え動作では食材を器具上で滑らせて落とすことで実現する。そのため、食材を器具上で滑らせるために必要となるパラメータを検討する必要がある。水気の多い食材は器具への張り付きや水気によって滑りやすくなることが考えられる。肉や魚のような柔軟な食材の場合には、力を加えたときに物体が変形し器具上で食材が滑りにくいことが考えられる。豆腐のような脆い食材では器具によって力を加えることで食材を変形・破壊してしまうことが考えられる。そこで、本稿では以下の4つのパラメータを用いて食材特性を表現し、ロボットによる特性に応じた動作を実現することとする。

- H:height[m] 様々な姿勢で取りうる最も低い高さ
- F:flexibility[-]: 柔軟性
- B:brittleness[-]: 脆さ
- W:wetness[-]: 水気

cleanは調理作業によって生じた作業領域の汚れを清掃するための作業である。直前に実行された調理作業と対象食材の組み合わせによって作業領域が濡れることや不衛生となることが考えられる。水気の多い食材や生の鶏肉や豚肉など不衛生なものを調理作業で使った後に、そのまま次の調理作業を行うと食材や作業に影響を及ぼすことが考えられる。よってそれらを清掃する作業が必要となる。



### 3.3 中間作業の生成

第2.1節で述べたように、中間作業は前後の調理作業の記述から導かれる。中間作業で必要な作業はそれぞれ必要となる条件が異なるため、前後の調理作業を比較して条件に適合する場合のみ実行されるべきである。直前の調理作業の対象食材と生成物をそれぞれ  $T_p, P_p$  とし、次の調理作業の対象と生成物をそれぞれ  $T_n, P_n$ 、対象食材を移動する作業領域の座標系を *workspace\_pose*、生成物を入れる容器を配置する座標系を *container\_pose* とする。またそれぞれの物体のメンバ属性を(物体).(属性の名前)の形で表す。

bringは前後の調理作業の対象食材と生成物の関係によって決定できる。以下にbringの実行条件と手順を示す。1.では対象食材の移動を実行する。前後の対象食材の容器が異なる場合に動作を実行する。2.と3.では生成物の移動を実行する。前後の生成物の移動先の容器が異なり、対象食材と生成物を入れる容器が異なる場合に、対象食材の移動を実行する。

1.  $T_p.container \neq T_n.container$  のとき,  
     $bring(T_p, T_p.initial\_pose)$   
     $bring(T_n, workspace\_pose)$
2.  $T_p.container \neq P_p.container$  and  $P_p.container \neq P_n.container$  のとき,  
     $bring(P_p, P_p.initial\_pose)$
3.  $T_n.container \neq P_n.container$  and  $P_p.container \neq P_n.container$  のとき,  
     $bring(P_n, container\_pose)$

collectでは直前の調理作業後に生成物を移動する必要がある、生成物が容器に入っていない場合に実行されるべきである。そこで、直前の調理作業の生成物の記述から、生成物を入れる容器を参照し、容器に生成物が収容されていない場合にcollectを実行する。collectでは容器に食材が収容されるため、 $P_p.container$ のメンバ属性のingredientに $P_p$ を登録する。

cleanは直前の調理作業で扱った食材が水気のある食材か、液体の場合に実行される必要がある。よって、 $T_p.character$ のwetnessが大きい場合と、 $T_p.type$ が液体である場合には実行されるべきである。しかし、作業領域に物体が存在する場合には清掃作業を実行

できない。そこで、bringの1.の条件下で、直前の対象食材を移動した場合にのみ、作業を実行することとした。

## 第4章

# 動作の実装

本章では，初めに提案手法の実装に使用したロボットシステムについて述べる．次に，第3章で述べた中間作業に必要な動作の実装について述べる．最後に先行研究の調理作業の再実装について述べる．

### 4.1 システム概要

#### 4.1.1 双碗ロボットシステム

図 4.1のように，三菱重工製7自由度の汎用ロボットPA-10を2機用いて双碗ロボットシステムを構築した．また，テーブル上での作業を実行しやすいように，ロボットは床面から約700mmの高さに配置した．図 4.2のように，左アームの台座の中心座標を原点として，ロボットから見て前方を $x$ 軸正方向，左側を $y$ 軸正方向，鉛直上方を $z$ 軸正方向として，ベース座標系を設定した．本論文では，座標系の姿勢を $x, y, z$ 軸の順に $\alpha, \beta, \gamma$ 回転させるオイラー角で表現する．右アームについても左アームのベースの中心を原点とした同じ位置・姿勢の座標系で制御を行う．以降，本稿では特に記述のない場合にはベース座標系に従って記述することとする．

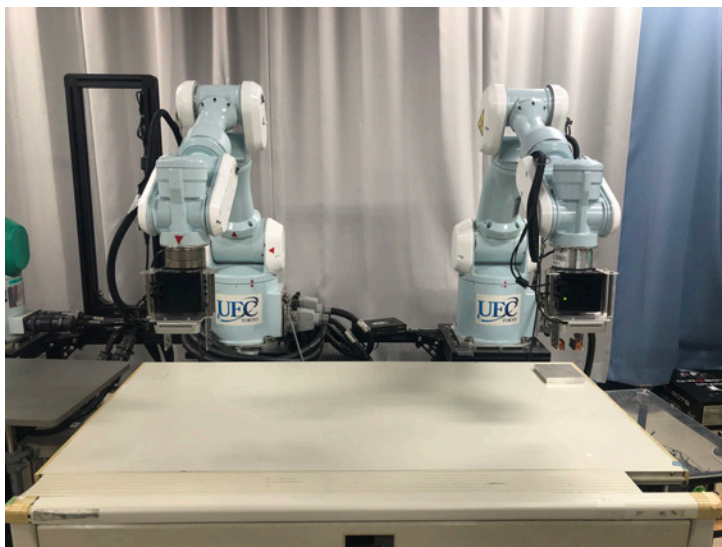


図 4.1: 双腕ロボットシステムの概観

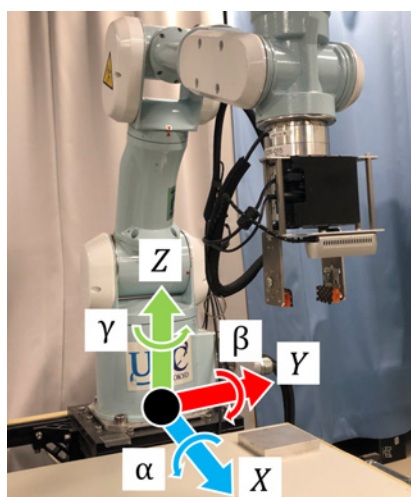


図 4.2: ロボットのベース座標系

#### 4.1.2 力覚センサ

両ロボットアームの手首関節の先にはLeprino製6軸力覚センサPFS080YA501U6Sを装着した。図 4.3にセンサの外観を示す。



図 4.3: 力覚センサ

### 4.1.3 平行二指ハンド

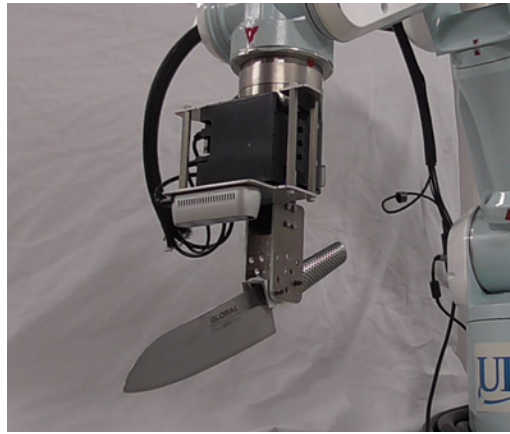
両ロボットアームの手先には三菱電機製の平行グリップ1E-HM01 HANDを装着した。料理作業では，食材や容器の把持に加えて，調理器具を把持しながら調理作業を行う。長野らによって開発されたハンド [10]は，アルミ板に多数の穴が設けられているため，様々な姿勢で安定してツールを装着可能である。第2章で述べたように，調理器具の把持は本研究の主旨とは異なるため，人の手による換装を行うこととする。

### 4.1.4 エンドエフェクタ

サラダを作成するために必要となる調理作業に使用する調理器具類のエンドエフェクタと，食材や容器を直接把持するためのエンドエフェクタを図 4.4,図 4.5,図 4.6,図 4.7に示す。調理器具類のエンドエフェクタは，市販されている一般的な調理器具と指先へ装着するためのアダプタで構成される。アダプタは調理器具を固定するためのアルミ板に安定して把持するためのダウエルピンをインターフェイスとして指先に装着される。食材や容器を把持するためのエンドエフェクタは，単純把持のために滑り止めシートを貼ったアルミ板をねじ止めすることで指先に装着する。



(a)包丁

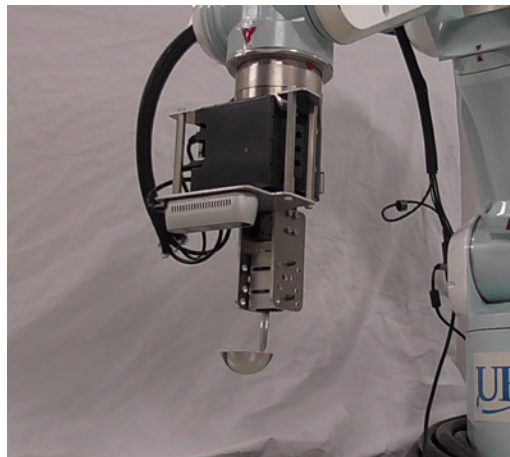


(b)ハンドに装着した包丁

図 4.4: 包丁

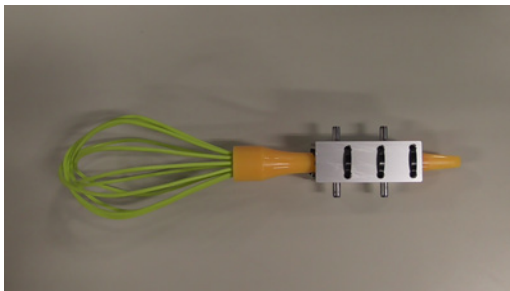


(a)スプーン

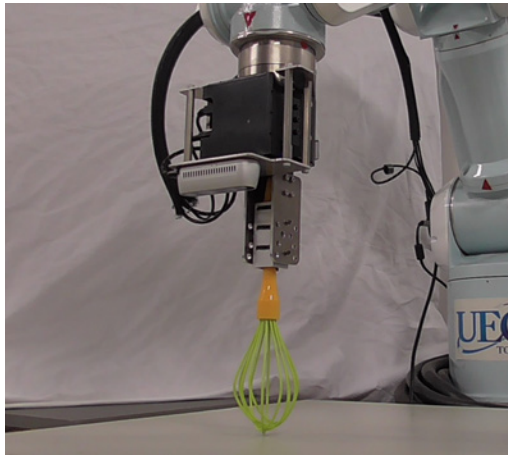


(b)ハンドに装着したスプーン

図 4.5: スプーン

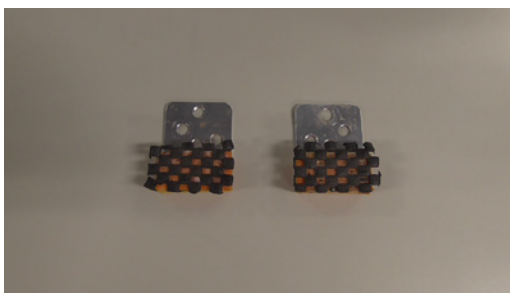


(a)泡だて器

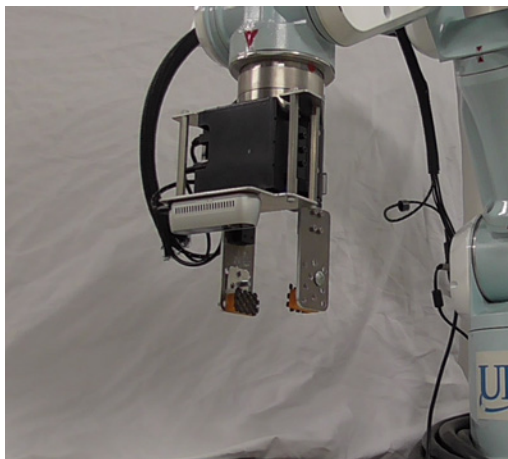


(b)ハンドに装着した泡だて器

図 4.6: 泡だて器



(a)物体把持エフェクタ



(b)ハンドに装着した物体把持エフェクタ

図 4.7: 物体把持エフェクタ

## 4.2 食材の配置

本節では、食材の配置作業について述べる。

第2章で検討した通り、複数の調理過程からなる料理作業では、ロボットの関節可動域の問題から作業領域と非作業領域間の食材移動が必要となる。

### 4.2.1 食材・容器の移動作業

3.2で述べたように食材・容器の移動作業はbringによって実行される。bringは移動に使用するアームや制御する座標系の決定と物体把持動作、物体の移動・解放動作で構成される。

双腕ロボットシステムでテーブル全体の物体の移動を行うためには、各アームの関節可動域による制限を考慮する必要がある。本研究では、それぞれのハンドの姿勢をテーブル平面に対して垂直にした状態で200mmの高さまで移動可能な範囲をアームの移動動作可能範囲 $R_{\text{left}}$ ,  $R_{\text{right}}$ として定めた。アームの寸法から順運動学によって求めた結果はベース座標を中心とした約845mmとなったが、実際に動作させてみると840mmでは関節が伸びきってしまったため、830mmを移動動作可能範囲として使用した。bringでは物体の移動元と移動先の把持位置がアームの移動動作可能範囲内の場合に限り、該当するアームで動作を実行することとした。また、両アーム共に実行可能な場合は左アームを使用することとした。

制御する座標系の決定は食材を格納する容器の有無によって決定される。容器に入っていない食材の移動は、3.1.1節で述べたingredientのgrasping\_coordinateとgrasp\_forceを利用してツール座標系を制御することで、食材を把持して食材移動を行う。食材に容器が紐づけられている場合は、containerのgrasping\_coordinateを用いて食材移動を行う。また、容器に食材が入っている場合には容器を持ち上げずに引きずって移動作業を行う。対して調理の過程でまだ食材が入っていない容器を移動するときは、容器の縁を把持して持ち上げてから容器の移動を行うこととする。

以下に移動作業のシステムの疑似コードを示す。bringでは食材の移動を行うため、食材と内包している容器の環境情報の更新が行われる。移動対象の食材のメンバ属性であるposeと容器に内包されている場合はその容器のposeを更新する。



---

**擬似コード 1 bring**

---

```
1: function bring(ingredient, destination)
2:   ingredient ← 移動する食材
3:   destination ← 移動先の座標
4:   Arm ← 使用するアーム
5:   Gripper ← 使用するアームのグリップ
6:   grasp_force ← 把持力
7:   if ingredient.position, destinationが $R_{\text{left}}$ の範囲内 then
8:     Arm ← Armleft
9:     Gripper ← Gripperleft
10:  else if ingredient.position, destinationが $R_{\text{right}}$ の範囲内 then
11:    Arm ← Armright
12:    Gripper ← Gripperright
13:  else
14:    エラーメッセージの表示
15:  grasping_coordinate ← 食材・容器の把持座標系
16:  if ingredient.containerが指定されている then
17:    grasping_coordinate ← ingredient.container.grasping_coordinate
18:  else
19:    grasping_coordinate ← ingredient.grasping_coordinate
20:    grasp_force ← ingredient.grasp_force
21:  Gripperの開放
22:  ツール座標系を把持座標系に移動
23:  Gripperを閉じて物体を把持
24:  if container.ingredientが指定されていない then
25:    200mm直上に移動してからArmを初期位置へ移動
26:  destinationにハンドを移動
27:  Gripperを開放して物体を置く
28:  Armを初期位置へ移動
29:  ingredient.poseを移動先の座標に更新
30:  ingredient.container.poseを移動先の座標に更新
31: end function
```

---

#### 4.2.2 食材集め・移し替え作業

本項では3.2で述べたcollectについて述べる。

本研究で食材移動に用いる器具は図 4.8のような正方形の板の二辺に傾斜がついている器具である。これを平行二指ハンドによって把持してまな板に押し付けながら、食材の付近に固定した包丁に向かって押し付けることによって器具上に食材を乗せる。そして、器具上の食材を容器に落とすことで食材の移動を達成する。

食材集め動作は以下の手順で実現する．

1. 食材位置から目標経路を決定する
2. 包丁を経路の終点に移動用器具を始点に構える
3. 包丁に器具を押し付けて食材をすくい上げる
4. 器具上の食材を包丁で寄せる

食材集め動作を行う際には目標となる食材座標系 $A_d$ ，移動用器具を制御するツール座標系 $A_{Tedge}$ ，包丁を制御するために刃先座標系 $A_{Ktip}$ を定義する．各座標系の定義を図 4.8 に示す．

目標経路は $A_D$ を中点として食材の幅に応じた $2L$ の直線とし，経路の両端には30mmの余裕を作る．これは食材と器具との接触を避けるためである．

包丁を終点からまな板に接触させて構えて，始点には移動用器具をツール座標系を用いて力 $F_1$ でまな板に押し付ける．そのままの姿勢で目標経路上を終点から30mmまで位置制御で移動させ，包丁との接触を検知するために力制御によって力 $F_2$ で押し付ける．器具の姿勢を机平面と平行にして，包丁を器具から高さ $H_K$ で器具の中央まで食材寄せ動作を行う．このとき，包丁による壁面への押し付けによって食材が破壊されないように，食材の脆さに応じて力 $F_3$ で制御する．以上のように食材寄せ動作を実現する．

各パラメータについては3.1.1項で述べた食材特性と実験結果から決定する．図 4.8(B)のように食材に器具のエッジを滑り込ませるために，机平面に対してエッジを $\theta$ 傾斜させて押し付ける．本稿では，きゅうりの乱切りを模した紙粘土製のモデル食材を使用した予備実験において， $\theta$ を $0^\circ$ から $20^\circ$ まで $10^\circ$ ずつ変化させたとき， $10^\circ$ で安定してすくい上げを実現できたため $\theta = 10^\circ$ とした．また，力制御を用いて $F_2 = 1.5\text{N}$ で包丁に器具を押し付けることで，安定して包丁との接触検知が可能となった．

$F_1$ は器具の変形や把持位置のズレを生じるため可能な限り小さくしたい．しかし，食材の水気で接触面がぬれて力が加わっている場合や食材の変形で器具の上に食材が乗りにくい場合には大きくする必要がある．そこで，特性パラメータの柔軟性 $F$ と水気 $W$ が大きいとき，力 $F_1$ を強くする．

包丁の高さ $H_K$ は，食材の高さが小さいときには寄せ残りが出ないように，力制御で器

具に接触させて寄せ動作を行う。しかし、包丁による接触によって、器具の変形や包丁の下に入り込んだ食材を切断してしまうことがある。そのため、食材に高さがある場合には一定の高さで寄せ動作を行う。

寄せ動作の力 $F_3$ は硬い食材のときは食材同士の接触により寄せ動作に強い力が必要な場合がある。しかし、脆い食材には強い力を加えると潰れてしまうため、力制御が必要となる。

以下に具体的な食材特性の条件式を示す。

$$F_1 = \begin{cases} 1[\text{N}] & (\text{flexibility} = 2 \text{ or } \text{wetness} = 2) \\ 0.7[\text{N}] & (\text{flexibility} = 1 \text{ or } \text{wetness} = 1) \\ 0.5[\text{N}] & (\text{otherwise}) \end{cases} \quad (4.2.1)$$

$$H_K = \begin{cases} 0[\text{mm}] & (\text{height} \leq 3[\text{mm}]) \\ 3[\text{mm}] & (\text{height} > 3[\text{mm}]) \end{cases} \quad (4.2.2)$$

$$F_3 = \begin{cases} 1[\text{N}] & (\text{brittleness} = 2) \\ 5[\text{N}] & (\text{otherwise}) \end{cases} \quad (4.2.3)$$

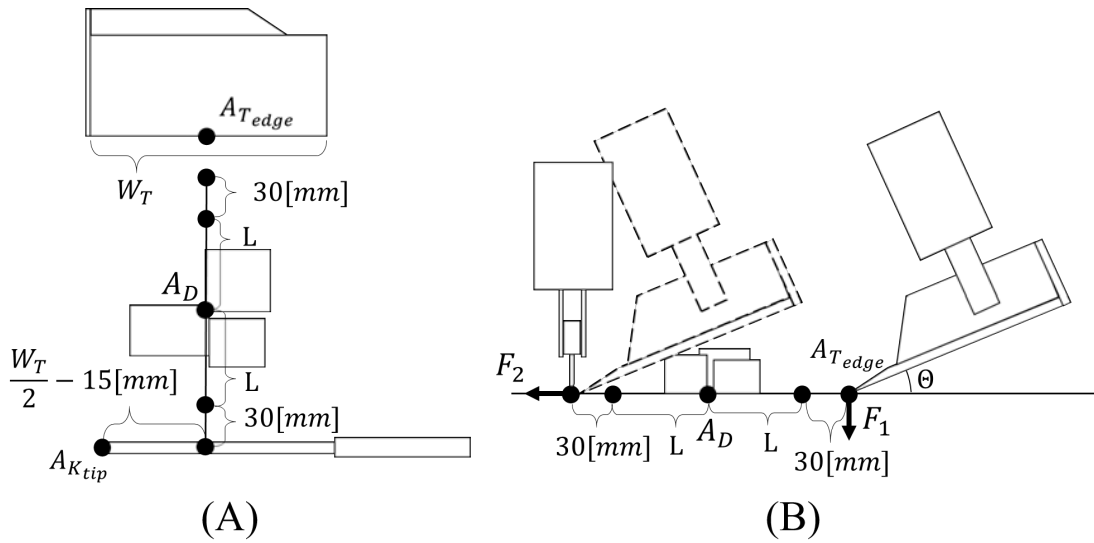


図 4.8: 食材集め動作の座標系とパラメータの定義

移し替え動作では食材を移動先の容器に移すために移動用器具から食材を落とす動作を実現する。動作は食材寄せ動作のように、包丁で食材を滑らせる方法と移動用器具を傾ける方法の2つを食材の特性に応じて選択する。

包丁で食材を滑らせる方法で用いる座標系とパラメータの定義を図 4.9に示す．この方法では，まずツール座標系 $A_{Tcorner}$ を用いて移動用器具を移動先の容器の上に図のように構える．図 4.9のように包丁を器具の端に接触させて構え，器具上を同じ姿勢で移動させることで，食材を移動先の容器に落とす．

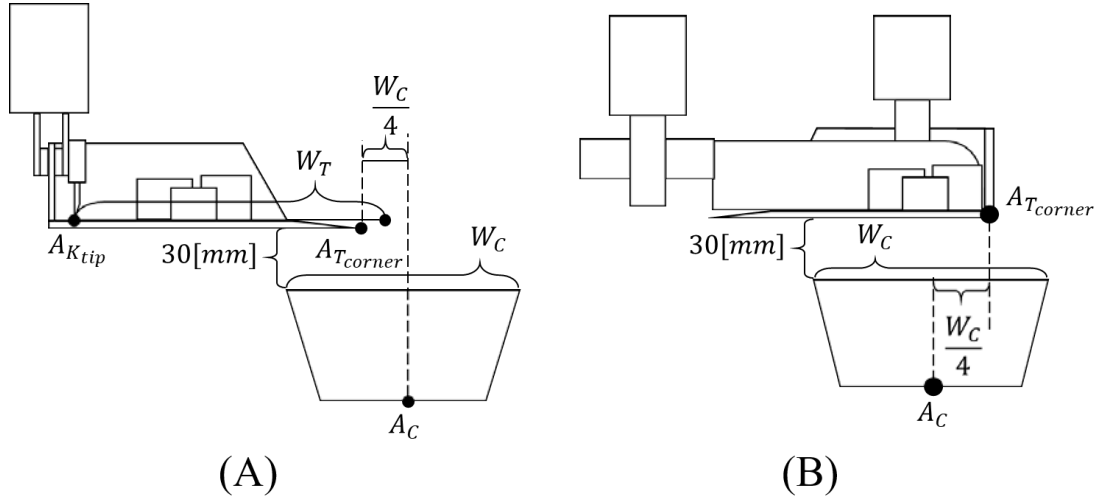


図 4.9: 移し替え動作の座標系とパラメータの定義

移動用器具を傾ける方法では，包丁で食材を滑らせる方法と同じ位置に移動用器具を構える．ただし，食材の意図しない方向への落下を防ぐため，移動用器具の姿勢を2つの壁面の方向に傾ける．その後ツール座標系 $A_{Tcorner}$ 上で容器側に傾けることで食材を容器に落とす．このとき，関節可動域の問題から壁面に傾ける角度は $15^\circ$ とし，容器に落とすための角度は $40^\circ$ と定めた．

包丁による移し替え動作では，包丁が食材に接触して切断してしまうことが考えられる．また，傾きによる移し替え動作は食材と器具の滑りが悪い場合に全く食材が動かないことや意図しない方向に落下することが考えられる．

そこで，食材の高さが極端に低い場合や非常に柔らかい場合には，重心の低さや変形により接触面が大きくなることが起因して器具を傾けても動かないことが考えられるため，包丁による移し替え動作のみを実行する．水気のある食材は器具に張り付きやすいため，同様に包丁による移し替え動作のみを実行する．水気の非常に多い食材では反対に器具が水でぬれ，滑りやすくなるため傾きによる移し替え動作を実行する．また，食材の高さが高い場合には傾け動作によって食材が落ちやすいため，包丁による移し替え動作を行わない．以下に食材特性に対する具体的な条件を示す．

- 移し替え動作のみ：  
 $\text{height} \leq 5 \text{ or flexibility} = 2 \text{ or wetness} = 1$
- 傾け動作のみ： $\text{height} > 35$
- 両方を実行： $\text{otherwise}$

### 4.3 環境の初期化

本節では，調理作業後の環境の初期化作業の実装について述べる．3.2で述べた通り，調理作業と食材の組み合わせによっては作業領域の清掃が必要となる場合がある．そこで，本研究ではロボットによる作業領域のふき取り動作を実行することで，作業領域の清掃する．図 4.10にふき取り動作の制御方法を示す．ハンドで布巾やキッチンペーパーを把持した状態で図のように走査的にふき取り動作を行うことで，作業領域全体の清掃を行う．水気の多い食材を調理作業で使用した後は作業環境が汚れ，次の調理作業に影響を及ぼす可能性があるため，水気 $W = 2$ ，または対象食材のtypeが液体である場合にcleanを行うこととした．

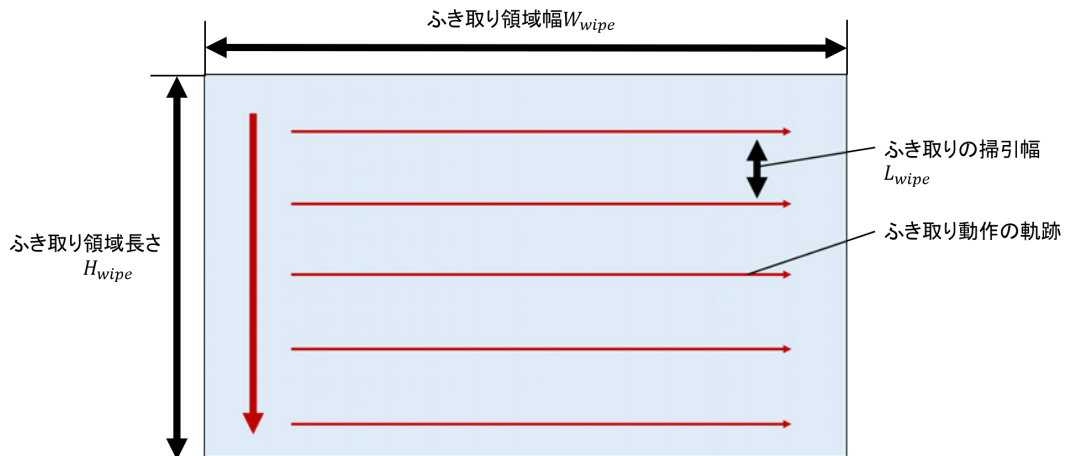


図 4.10: ふき取り動作

## 4.4 調理作業の実装

本節では、サラダの作成に用いる切断・計量・混合作業について考察し、ロボットへの実装方法を検討する。これらの作業はそれぞれ調理作業の研究として提案されて実装されている。そのため、本研究ではこれらの手法のパラメータとシステムを再検討して使用することとする。パラメータの検討には5章のサラダ作成実験と同じ調理器具を用いた。

再利用性が高く一般化された切断作業は、今井らが提案し実装している [5]。実装された切断動作に必要なパラメータは、切断時の力覚フィードバックのゲインパラメータと力の目標値、突き切り時のストローク幅、まな板の接触を検知する力覚の閾値である。本研究では、食材モデルの属性を用いることで、食材ごとに適切なパラメータを設定する。

本研究ではトマトの切断作業が必要となるため、トマトを切断するのに適切なパラメータを実験によって経験的に導出した。ゲインパラメータはフィードバックの値に発振等が見られなかったため、同じパラメータを使用した。力の目標値は5-20Nまで5N刻みで切断実験を行ったところ、10Nで安定して切断可能であることがわかったため、10Nと設定した。まな板の接触を検知する力の閾値は目標値よりも十分高い値であれば良いため、20Nと設定した。突き切りのストローク幅は包丁と食材の大きさから40mmと定めたまた、トマトは水分の多い食材であるため、切断時に刃に切断した食材が張り付き、食材が散らばる課題が発生した。そこで、力覚フィードバックによってまな板との接触が確認出来たときに、包丁の刃を軸にして10°回転させる動作を入れることで、安定して切り離すことができた。

計量作業は福島らが提案、実装を行っている [8]。本研究ではドレッシングの作成のために液体計量作業が必要となるため、液体計量作業に関するパラメータの再検討と再実装を行った。ドレッシングの食材にはデブスカメラによるセンシングの難しい酢を設定したため、力覚による液面検知を行った。実験の結果、力覚フィードバックの閾値を0.15Nとすることで、安定してスプーンの反力を検知できた。醤油についても同様の結果が得られたため、同じ値を用いることとした。また、計量作業では食材の移動を伴う。よって、生成物の容器`product.container`のメンバ属性である`ingredient`に`target`を登録する。

混合作業は松田らが提案、実装を行っている [11]。本研究では、ドレッシングの作成に液体混合作業が必要となるため、再実装を行った。

## 第5章

# 実験

### 5.1 切断済み食材の移動実験

#### 5.1.1 実験方法

対象食材と使用する容器を並べ、それらの情報を環境データとして入力する。本稿では対象食材の認識は行わず、目標の位置と座標系を既知のものとして与えて実験を行った。また、ロボットによる調理作業は行わず、まな板に調理作業後の環境の想定として、人の手によって切る調理作業を行った後の作業領域の環境を作る。これに対して食材集め動作・移し替え動作を実行する。

図 5.1におけるまな板の中央に食材を置いて食材集め動作を行う。なお、実験ではアームの可動域とアーム同士の干渉を考慮して、右手で手前に包丁を構えて左手で $x$ 軸の負の方向へ器具を移動させる目標経路を設定した。

また、実験の対象食材は以下の8種類とした。それぞれの食材に定義したcharacterを表 5.1に示す。選定理由としては、料理の切る作業で頻繁に用いられる野菜・果実類、肉や魚、豆腐やチーズといった加工食品の中から、硬さや水気などの特性が異なる食材を選定した。また、切り方については頻繁に行われるものの中で大きさや形状が異なる方法を選定した。

- きゅうり（小口切り1-2[mm]厚，乱切り）
- ニンジン（輪切り3-4[mm]厚，乱切り）
- トマト（くし形切り）
- 玉ねぎ（みじん切り）
- 豆腐（一口大）
- 鶏肉（ぶつ切り）

表 5.1: 実験に用いた食材特性

食材	状態	H[mm]	F[-]	B[-]	W[-]
きゅうり	小口切り	1	1	0	1
	乱切り	21	0	0	1
にんじん	輪切り	2	0	0	0
	乱切り	28	0	0	0
トマト	くし形切り	23	1	1	2
玉ねぎ	みじん切り	1	1	0	1
とうふ	一口大	19	1	2	2
鶏肉	ぶつ切り	15	2	0	1



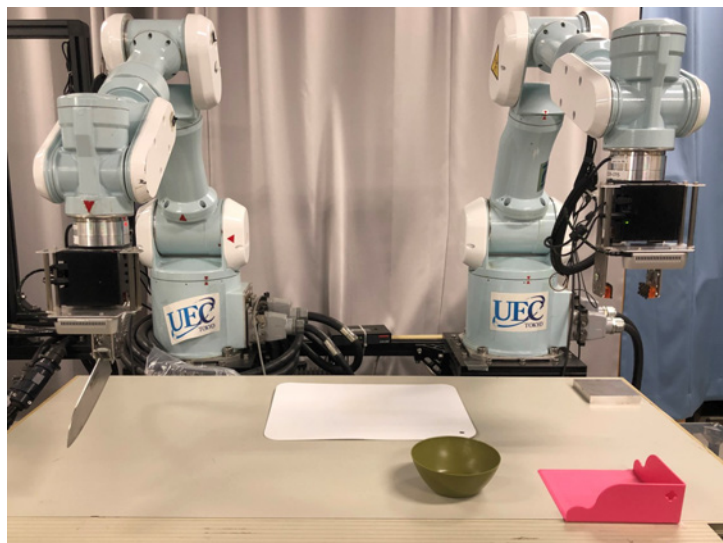


図 5.1: 実験環境



図 5.2: 使用器具

### 5.1.2 結果・考察

結果を表 5.2に示す．ここで，○は移動成功，残りは移動元に食材が残ってしまった場合，零れは移動先の容器に入らずに食材が落ちてしまった場合を表す．また，にんじんの乱切りの実験過程を図 5.4に示す．

にんじんの乱切り，トマトのくし形切り，壊れやすい豆腐や柔らかい鶏肉について提案した手法で移動作業を実現できた．輪切り・乱切り・みじん切りなど食材の形状が極端に薄く小さいものは，包丁による押さえや食材を寄せる動作の際に，移動用器具の変形による隙間や食材同士の押し出しが生じ，食材の集め残しや食材寄せの失敗が発生した．移し替え動作についても同様の理由から食材の残りが生じた．きゅうりの乱切りについては，移し替え動作の際に包丁がきゅうりと接触して移動用器具が変形し，そのまま落とし動作

を行ったため、食材の残りが生じた。

食材集め動作における食材残りは、視覚情報による残り食材の認識により食材の移動作業を繰り返し行うことで改善できると考えられる。きゅうりの乱切りにおける移し替え動作の食材残りは、包丁を近づけて食材との接触が発生した際に、再度接触が起こらないように食材寄せ動作を行うことで解決できると考えられる。薄い食材については刃が直線形状の包丁を使用して、移動用器具との隙間を作らないなどの改善が求められる。

表 5.2: 実験結果

食材	食材集め動作	移し替え動作
きゅうり(小口切り)	残り	残り・零れ
きゅうり(乱切り)	○	残り・零れ
にんじん(輪切り)	残り	零れ
にんじん(乱切り)	○	○
トマト(くし形)	○	○
玉ねぎ(みじん切り)	残り	残り・零れ
豆腐(一口大)	○	○
鶏肉(ぶつ切り)	○	○

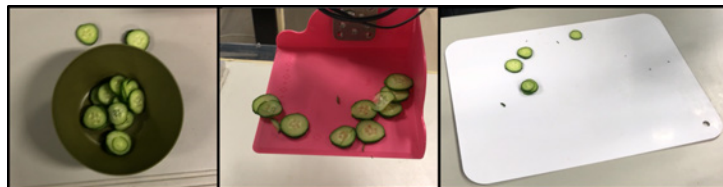
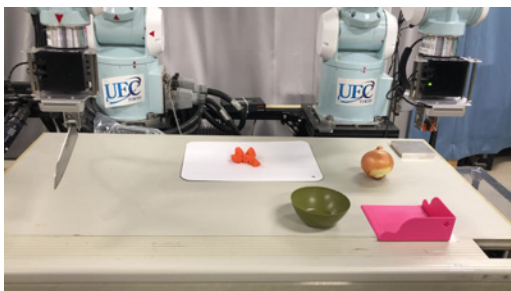
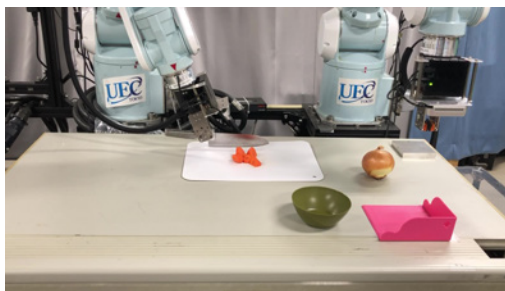


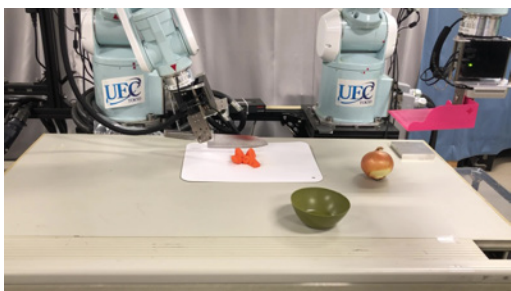
図 5.3: きゅうりの小口切りの結果



(a)作業開始



(b)包丁を構える



(c)移動用器具の把持



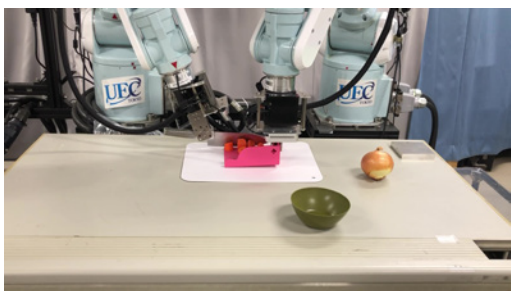
(d)器具を始点の上部に移動



(e)まな板に力制御で押し付ける



(f)包丁に力制御で押し付ける



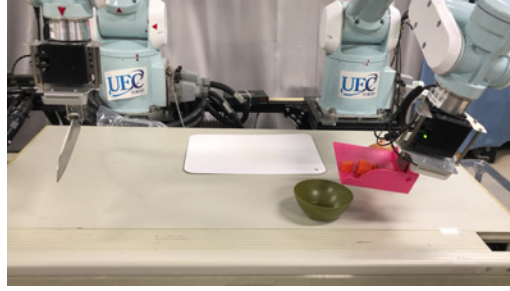
(g)器具の姿勢を平行に戻す



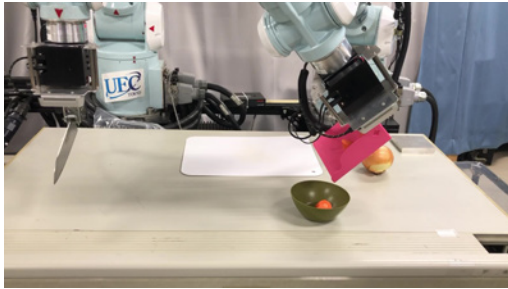
(h)包丁で食材を中央に寄せる



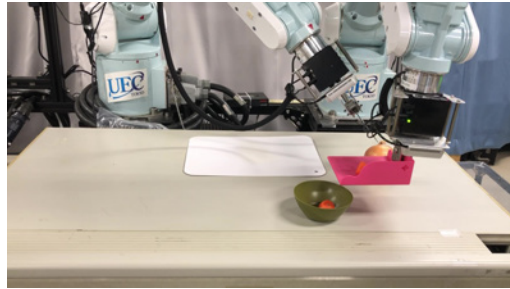
(i)食材集め完了



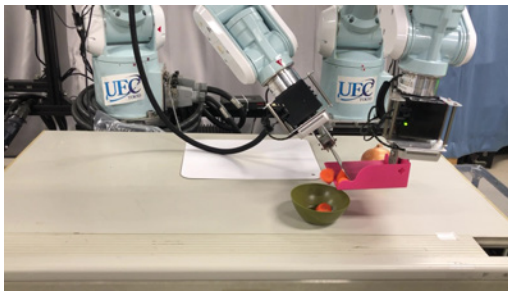
(j)容器上部へ移動



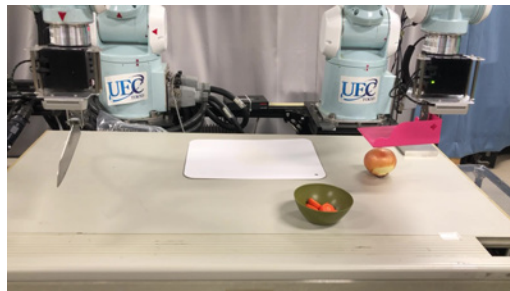
(k)移し替え動作を実行



(l)包丁を器具の端に構える



(m)包丁による移し替え動作を実行



(n)移動作業完了

図 5.4: 切断済み食材の移動実験の動作過程

## 5.2 サラダ作成実験

本節では、提案した中間作業と複数の調理作業の組み合わせによってサラダを作成する実験を行う。

### 5.2.1 実験方法

本実験では、簡易なサラダのレシピとして以下のようなものを用いる。パラメータの調節により、切断・計量・混合作業による調理作業を追加することは可能であるが、実行する中間作業には大きく影響しないため、レシピは簡易なものを用いる。

1. トマトを輪切りにする
2. 酢と醤油を3:1で量る
3. 2.を混合する
4. 3.を1.に量り入れる

図 5.5に作業実行前の環境を示す。料理作業で使用する容器を図 5.7, 図 5.8に示す。料理に使用する容器は実際の家庭環境に近いものを使用するため、大創産業製のポリプロピレンのボウル2種と保存容器を用いた。調理作業に使用する調理器具類は第4.1.4項で示したものを使用する。各調理器具には平行二指ハンドで安定して把持できるように治具を装着している。清掃作業にはキッチンペーパーを用いた。ロボットが使用する調理器具類は人の手で換装することとする。また、トマトのヘタを切り取る調理作業は実装されていないため、図 5.6のようにあらかじめヘタを取ることで対応した。

本実験では容器や食材の位置といった環境情報や、把持に用いる座標系は既知のものとして与える。本実験で入力した食材、容器、調理器具に関する情報の中で、作業環境に依らないレシピに関する項目を表 5.3, 表 5.4, 表 5.5に示す。サラダ作成の動作手順に基づく料理スクリプトを表 5.6に示す。



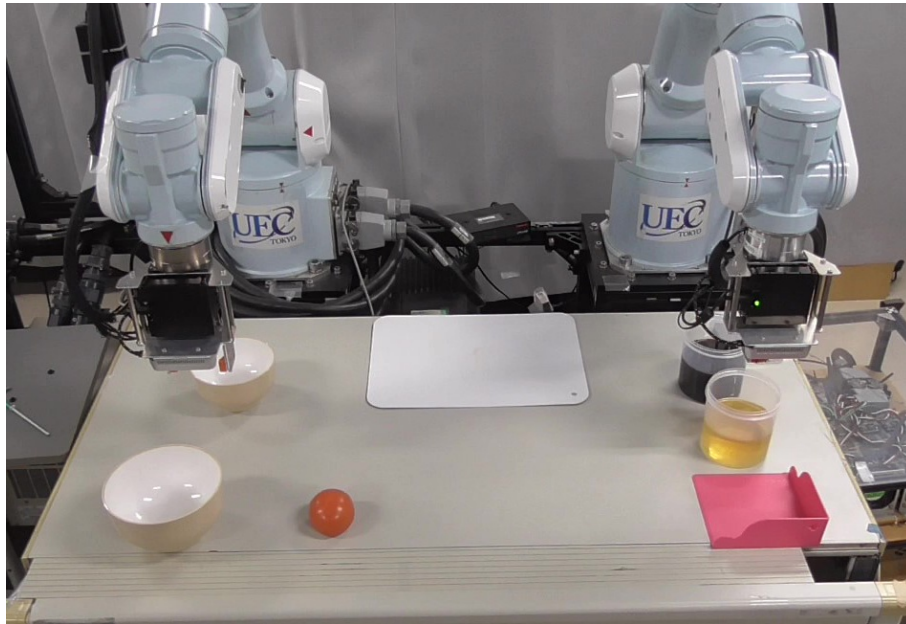


図 5.5: 実験環境



図 5.6: トマトの事前処理



図 5.7: サラダボウル(左)とドレッシング用のボウル(右)



図 5.8: 醤油と酢の容器

表 5.3: レシピに関する食材情報

ingredient	name	container
ings[0]	tomato	
ings[1]	vinegar	containers[1]
ings[2]	soy_source	containers[2]
ings[3]	dressing	containers[3]
ings[4]	salad	containers[4]

表 5.4: レシピに関する容器情報

container	name	ingredient
containers[0]	board	
containers[1]	soup_storage	ings[1]
containers[2]	soup_storage	ings[2]
containers[3]	soup_bowl	
containers[4]	salad_bowl	

表 5.5: レシピに関する調理器具情報

tool	name
tools[0]	knife
tools[1]	whisk
tools[2]	laddle_spoon



表 5.6: 入力した料理スクリプト

direction	target	product	tool	cook_direction	
				name	func
dirs[0]	ings[0]	ings[4]	tools[0]	輪切り	cut
dirs[1]	ings[1]	ings[3]	tools[2]	液体計量	measure
dirs[2]	ings[2]	ings[3]	tools[2]	液体計量	measure
dirs[3]	ings[3]	ings[3]	tools[1]	液体混合	mix
dirs[4]	ings[3]	ings[4]	tools[2]	液体計量	measure

### 5.2.2 結果・考察

図 5.12, 図 5.13, 図 5.14, 図 5.15, にサラダ作成実験の実験過程を示す. 図 5.10, 図 5.11には作成したドレッシングとサラダを示す. また, 図 5.9に清掃作業に使用したキッチンペーパーを示す.

図 5.12からトマトを切断し, 目標となる容器に切断したトマトを移動して, 作業領域の清掃を実行できていることがわかる. また, 図 5.9から調理作業後の作業領域の汚れのふき取りが出来ていることがわかる. これにより, 連続する調理作業における作業間の影響を抑えることができると考えられる.

図 5.13, 図 5.14から醤油と酢を量り混合することで目標となる容器にドレッシングを作成出来ていることがわかる. そして, 図 5.15のように作成したドレッシングをトマトにかけて一連の料理作業を達成した. 結果として, 図 5.11のように簡易なサラダが作成できたことが確認できる.

実験結果から, 提案した中間作業と料理スクリプトによる作業実行システムによって, 固体や液体など様々な特性を持ち調理作業により特性の変化していく食材に対して, 料理作業を実現することができると考えられる. しかし, 作業領域の汚れを一部取り損ねるなどの課題も残った. 本実験では画像認識を用いず, 入力した環境情報のみを用いて作業を実行している. そのため, 画像認識を用いることで, リアルタイムな視覚フィードバックによる動作計画ができるため課題を改善できると考えられる.



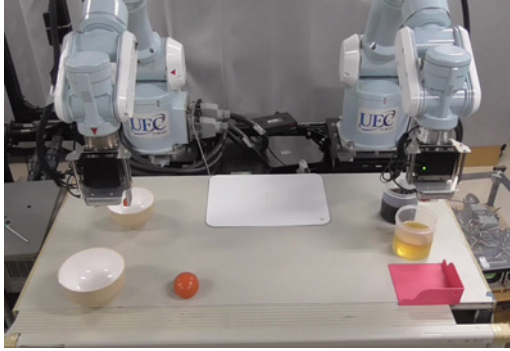
図 5.9: 清掃作業後のキッチンペーパー



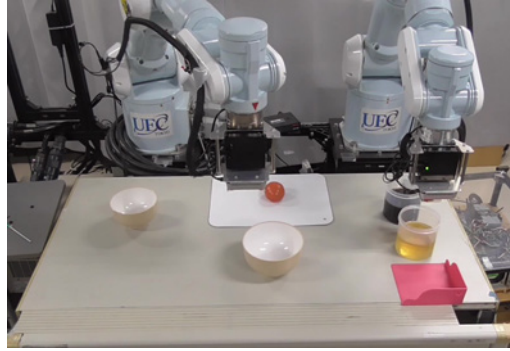
図 5.10: ドレッシングの作成結果



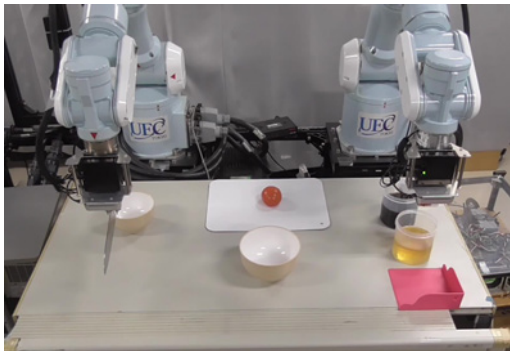
図 5.11: サラダの作成結果



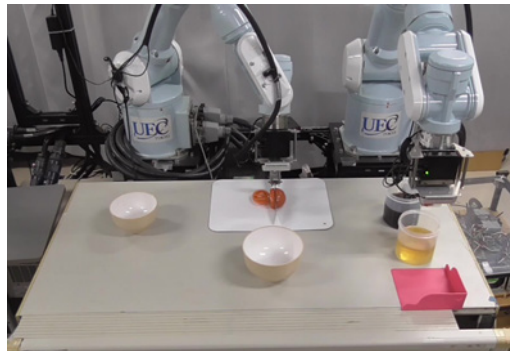
(a)料理作業開始



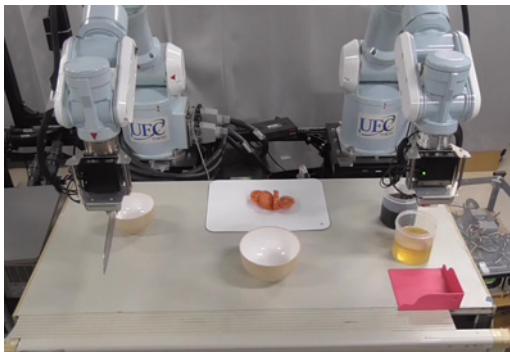
(b)食材と容器の移動



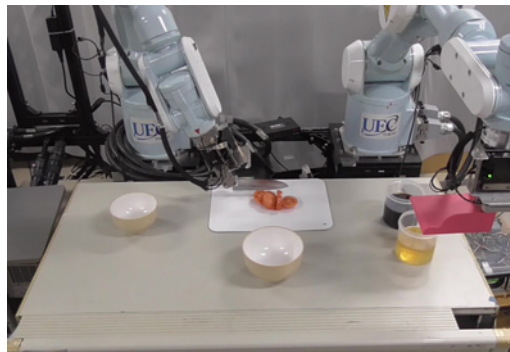
(c)人の手による器具の換装



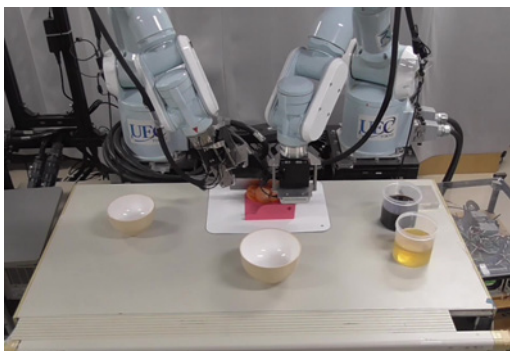
(d)切断作業の実行過程



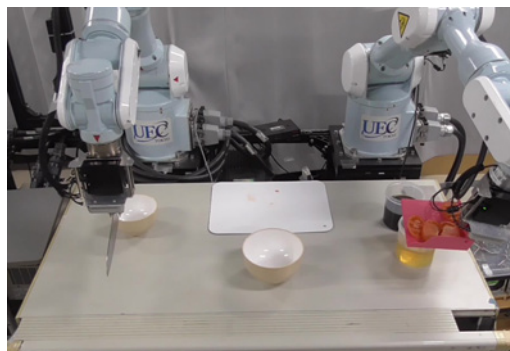
(e)切断作業の結果



(f)調理済み食材の移動作業開始

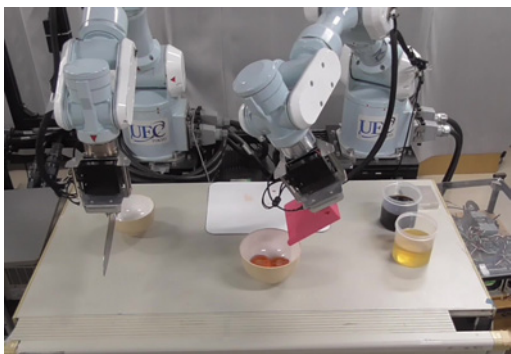


(g)食材集め動作の実行過程

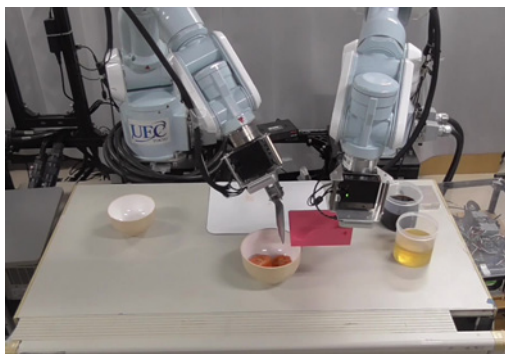


(h)食材集め動作の結果

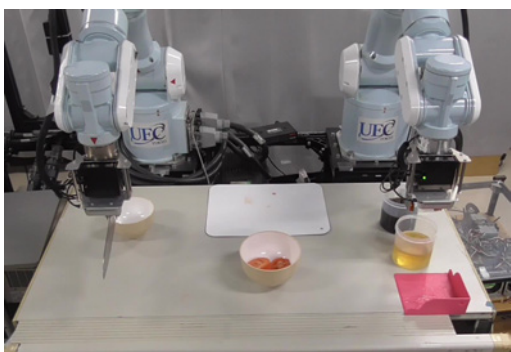




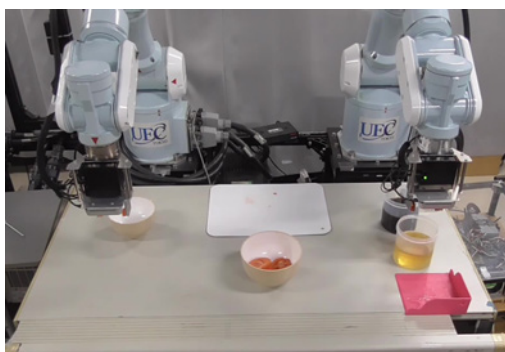
(i) 移し替え動作の実行



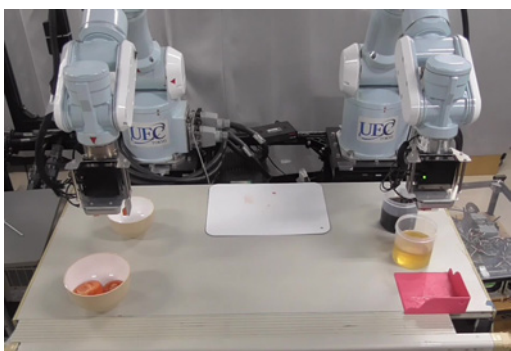
(j) 包丁による移し替え動作の実行



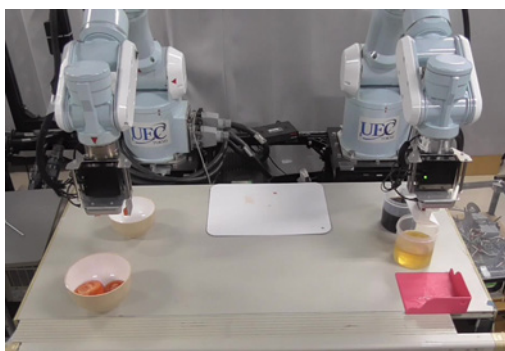
(k) 食材移動作業の実行結果



(l) 人の手による器具の換装



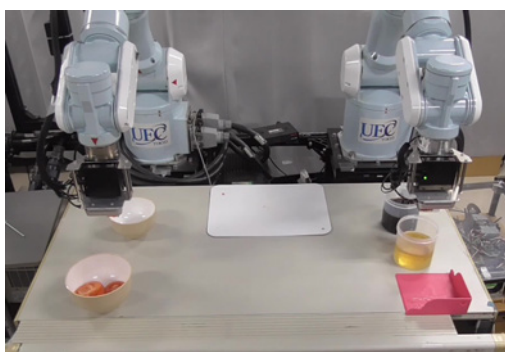
(m) 食材と容器の移動



(n) 人の手による器具の換装

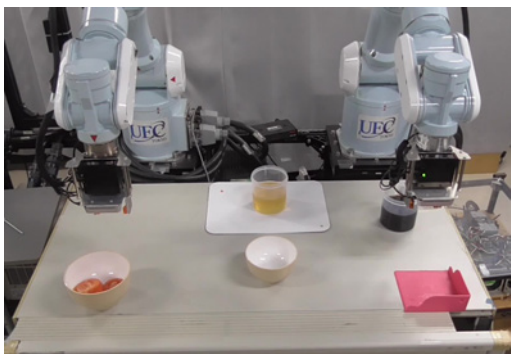


(o) 初期化作業の実行

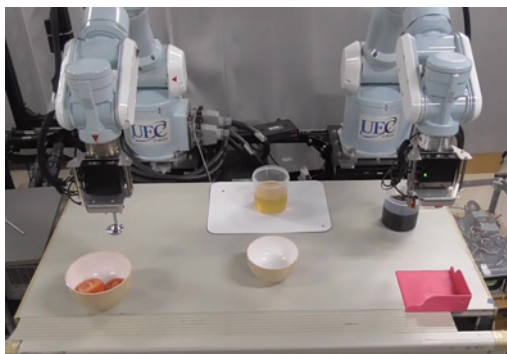


(p) 人の手による器具の換装

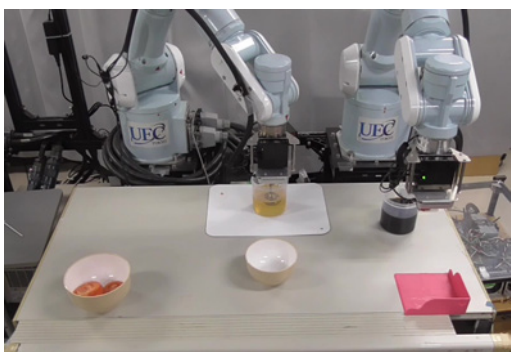
図 5.12: サラダ作成実験(トマトの切断作業)



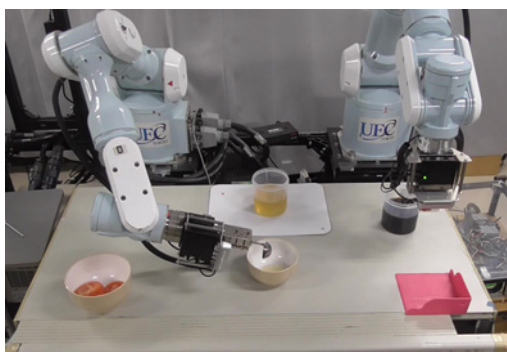
(a)食材と容器の移動作業



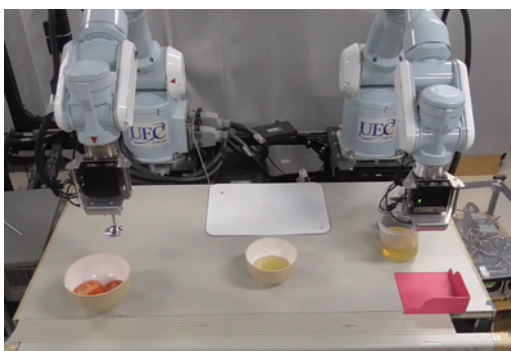
(b)人の手による器具の換装



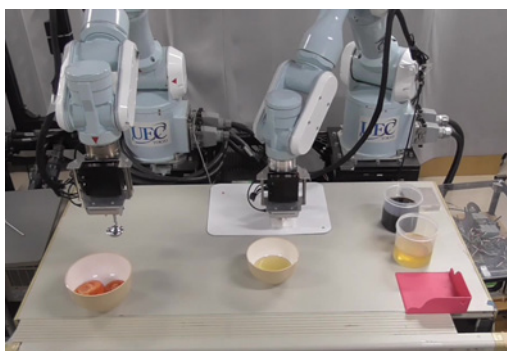
(c)計量作業の実行過程



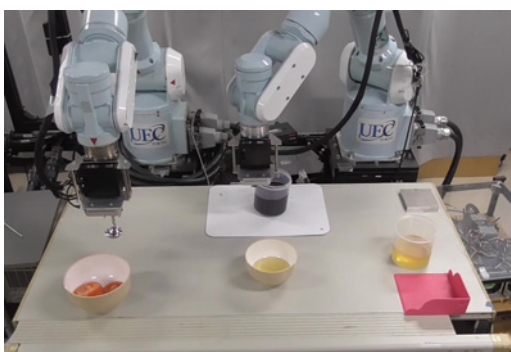
(d)計量作業の実行過程



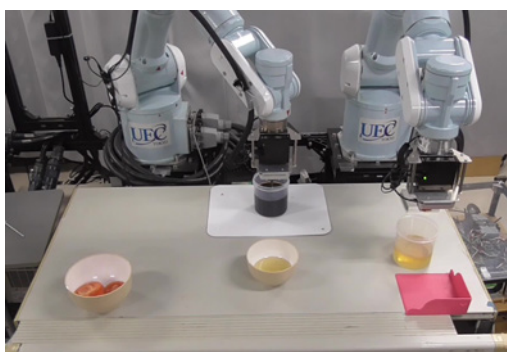
(e)計量作業の実行結果



(f)初期化作業の実行

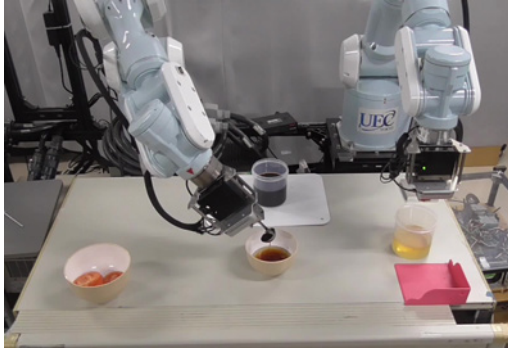


(g)食材と容器の移動作業

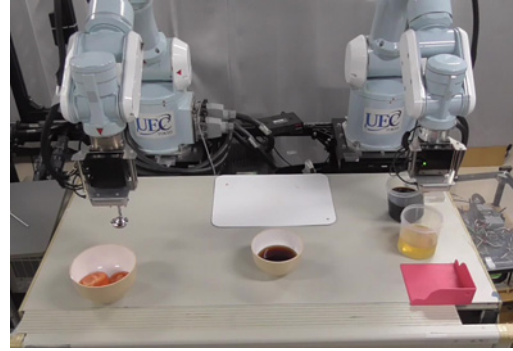


(h)計量作業の実行過程

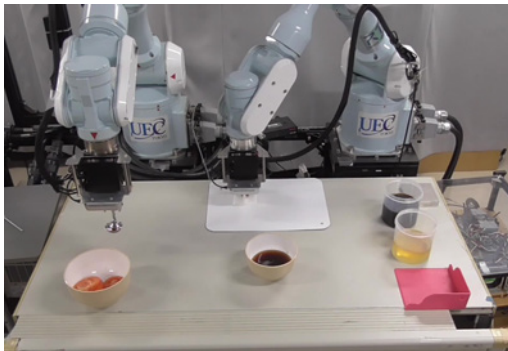




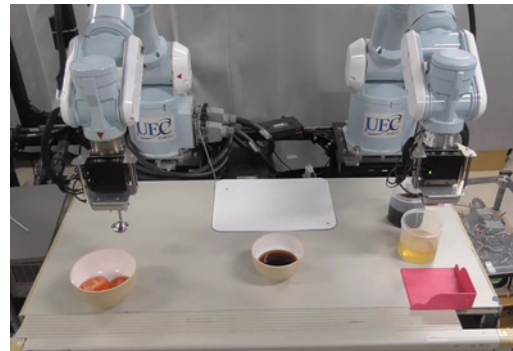
(i)計量作業の実行過程



(j)計量作業の実行結果

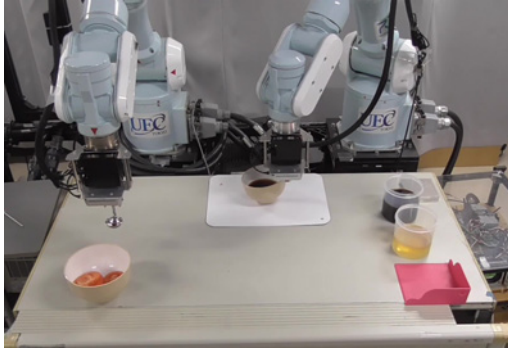


(k)初期化作業の実行

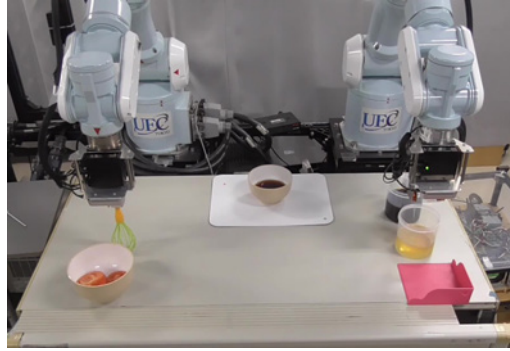


(l)計量作業の終了

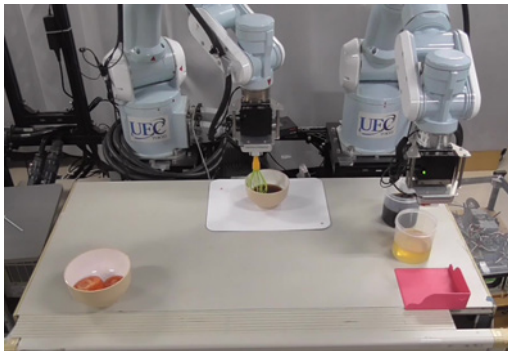
図 5.13: サラダ作成実験(酢と醤油の計量作業)



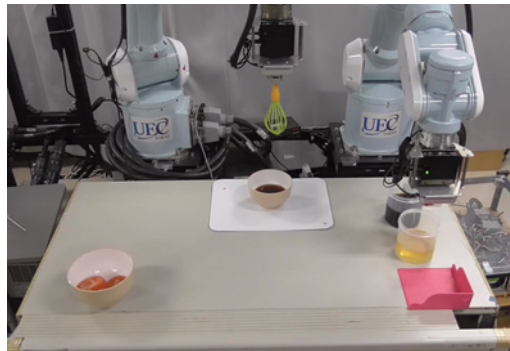
(a)食材の移動



(b)人の手による器具の換装

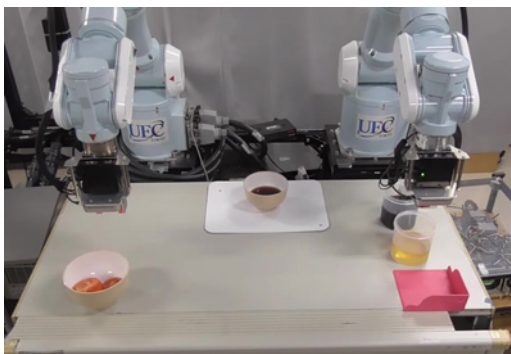


(c)混合作業の実行過程

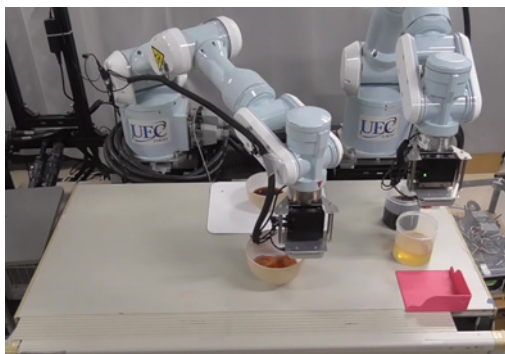


(d)混合作業の実行結果

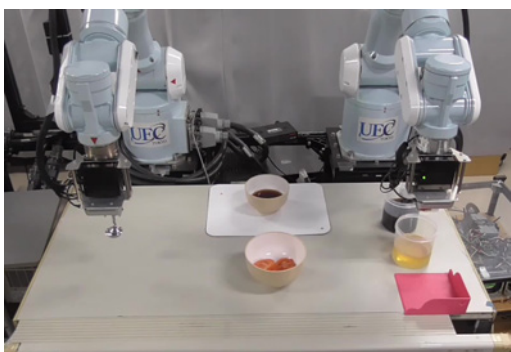
図 5.14: サラダ作成実験(ドレッシングの混合作業)



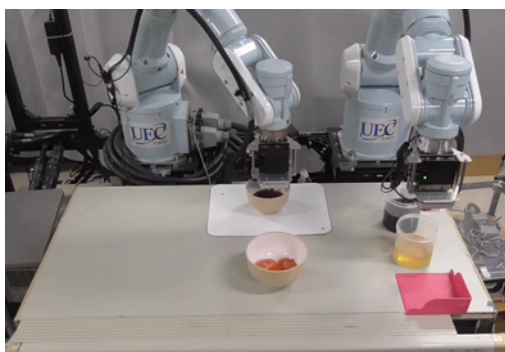
(a) 人の手による器具の換装



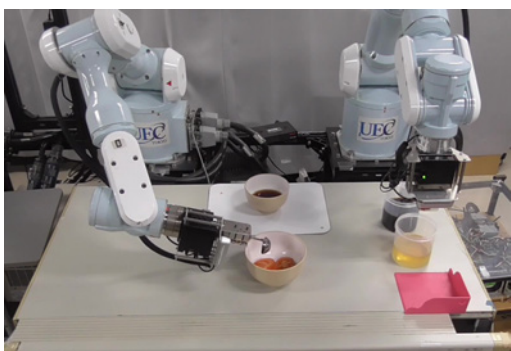
(b) 食材の移動



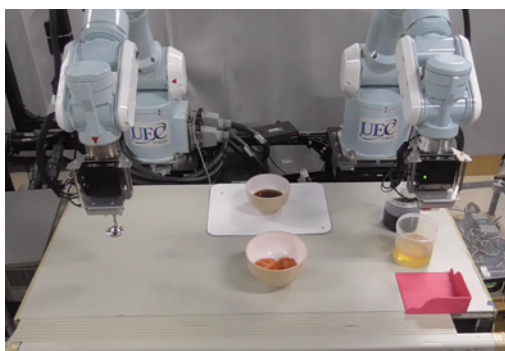
(c) 人の手による器具の換装



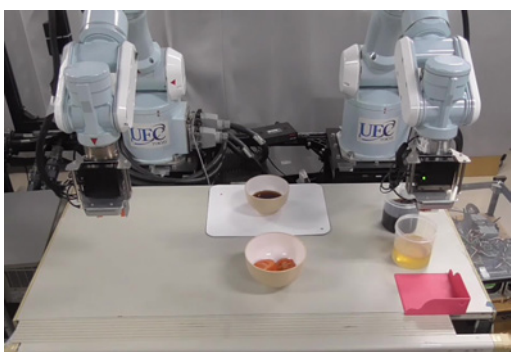
(d) 計量作業の実行過程



(e) 計量作業の実行過程



(f) 計量作業の実行結果

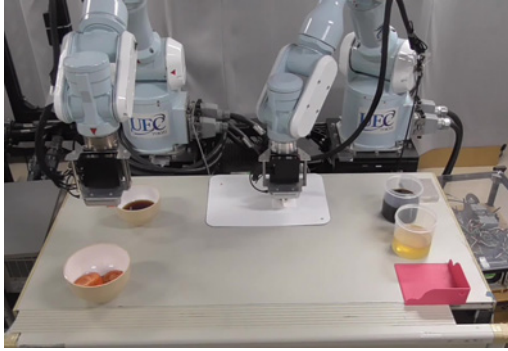


(g) 人の手による器具の換装

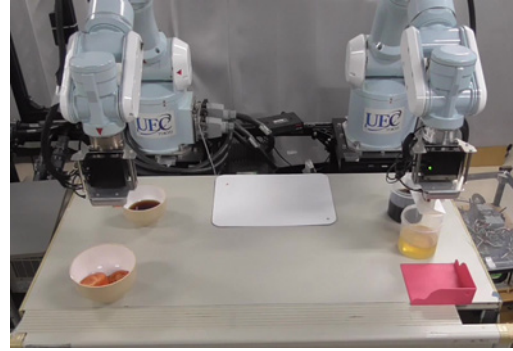


(h) 食材の移動





(i)初期化作業の実行



(j)料理作業終了

図 5.15: サラダ作成実験(ドレッシングの計量作業)

## 第6章

# 結言

### 6.1 まとめ

本研究では、双碗ロボットシステムで料理作業を行うことを主題として定め、複数の調理作業からなる料理作業を実現するために、調理作業間で前提とされている中間作業を抽出し、一連の料理作業を実行するシステムを構築することを研究目的とした。これを達成するために、以下のような課題に取り組んだ。

1. 調理作業間に必要となる中間作業の検討
2. 料理作業を具体的な動作に落とし込む記述の検討
3. 必要な動作と作業の記述を実行するシステムの実装

1. では先行研究で実現されている調理作業における作業の開始時点と終了時点の前提条件から、必要となる中間作業を抽出した。ここで、調理作業は非常に多岐に渡るため、具体的な課題として設定したサラダの作成に必要な調理作業である切断・計量・混合作業を軸に中間作業の検討を行った。2. では料理作業を単一の調理作業の連続として表現することで、調理作業間に必要な作業を抽出し、具体的な動作に落とし込むための記述の検討を行った。3. では1., 2. で検討した作業を双碗ロボットシステム上で実現するために必要な動作とそれらを実行するシステムの実装を行った。また、提案手法の検証を行うために、サラダ作成のための料理スクリプトを設定し、必要な調理作業の再実装を行って、適切な中間作業を生成することでサラダの作成を実現できた。

## 6.2 今後の展望

今後は本手法がどの程度複雑な作業まで適応できるのかを明らかにすることが課題として挙げられる。本研究における中間作業と作業記述は、サラダの作成に必要な切断・計量・混合作業を軸に検討・実装を行った。これらの作業をさらに多くの回数実行することは現在のシステムでも可能であると考えるが、より複雑な調理作業を含めた料理作業には限界があると考え、本研究で実装した中間作業の動作を他の調理作業の前提条件の処理に適用することができるか検討する必要がある。

また、本研究では各調理作業の前提条件のひとつである調理器具の把持は、中間作業の研究という目的からそれるため人の手による換装で行った。しかし、実際に双腕ロボットによる自律的な料理作業を実現するためには、調理器具の把持は欠かすことのできない課題である。調理器具の安定した把持と様々な食材の把持が可能となるエンドエフェクタなどを開発する必要がある。

# 謝辞

本研究を進めるにあたり，日頃から多くのご助言やご指導を頂いた末廣尚士教授，工藤俊亮准教授に深く感謝の意を表します．また，研究班の先輩としてご指導をいただきました董晨宇さん，滝澤優さんに感謝の意を表します．

最後になりますが，研究や日々の学生生活など様々な場面でお世話になった末廣・工藤研究室の皆様のおかげで充実した研究室生活を送ることができました．ありがとうございました．

# 参考文献

- [1] 総務省. 令和2年版情報通信白書. <https://www.soumu.go.jp/johotsusintokei/whitepaper/ja/r02/pdf/n1400000.pdf>.
- [2] 渡辺義明, 長濱虎太郎, 山崎公俊, 岡田慧, 稲葉雅幸. 2a2-b18 力覚状態認識に基づく包丁を用いた野菜切断行動のヒューマノイドによる実現. ロボティクス・メカトロニクス講演会講演概要集, Vol. 2010, pp. 2A2-B18.1-2A2-B18.4, 2010.
- [3] Connor Schenck and Dieter Fox. Visual closed-loop control for pouring liquids, 2016.
- [4] 吉田尚史, 吉見卓, 水川真, 安藤吉伸. ロボットアームによる調理動作の研究－かき混ぜ動作－. 第15回計測自動制御学会システムインテグレーション部門講演会 (SI2014) , pp. 1482-1485, 2014.
- [5] 今井啓明. 双腕ロボットによる料理作業実現のための食材の切断に関する研究～抽象操作記述と対象認識に基づく動的軌道生成～. 電気通信大学IS科MS専攻2015年度修士論文, 2015.
- [6] C. Dong, L. Yu, M. Takizawa, S. Kudoh, and T. Suehiro. Food peeling method for dual-arm cooking robot. In *2021 IEEE/SICE International Symposium on System Integration (SII)*, pp. 801-806, 2021.
- [7] F. Gravot, A. Haneda, K. Okada, and M. Inaba. Cooking for humanoid robot, a task that needs symbolic and geometric reasonings. In *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, pp. 462-467, 2006.
- [8] 福島洋. 食材を計量しドレッシングを作成する料理ロボットの実現. 電気通信大学IS科MS専攻2016年度修士論文, 2016.
- [9] 令和元年国民健康・栄養調査報告. <https://www.mhlw.go.jp/content/>

000711004.pdf.

- [10] 長野恵典. 双腕ロボットによるサンドイッチの作成に関する研究. 電気通信大学情報理工学研究科情報学専攻2017年度修士論文, 2017.
- [11] 松田啓明, 今井啓明, 工藤俊亮, 末廣尚士. ロボット用料理レシピに基づいた混ぜ操作の提案—粉体・液体・ペース状物体の混ぜ操作—. 第16回計測自動制御学会システムインテグレーション部門講演会 (SI2015) , pp. 2636–2641, 2015.